NOVA IMS Information Management School

MAA

Mestrado em Métodos Analíticos Avançados
Master Program in Advanced Analytics

Using Machine Learning to Predict Mobility Improvement of Patients after Therapy

A Case Study on Rare Diseases

Lara Barradas Teixeira Garrucho de Oliveira

Dissertation report presented as partial requirement for obtaining the Master's degree in Data Science and Advanced Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

Using Machine Learning to Predict Mobility Improvement of Patients after Therapy

by

Lara Barradas Teixeira Garrucho de Oliveira

Dissertation presented as partial requirement for obtaining the Master's degree in

Data Science and Advanced Analytics, specialization in Data Science

Advisor: Leonardo Vanneschi

Co Advisor*:* Illya Bakurov

July 2022

AKNOWLEDGES

I would like to thank all my family, friends, and people involved in this process. This would not be possible without your support.

Words cannot express my gratitude to Professor Carina Albuquerque and Professor Roberto Henriques, who taught me many valuable lessons and helped me grow professionally and personally during these years.

I am also grateful for the support of my supervisors, Leonardo Vanneschi and Illya Bakurov, who promptly advised and indicated to me the best steps to follow.

ABSTRACT

A disease is considered rare when it affects less than 5 people in 10 000 in the European Union. Although each disease individually affects a low number of patients, it is estimated to exist approximately 350 million people worldwide struggling with rare diseases. Despite these conditions threaten a significant part of the population, the cause, treatment and possible cure for many of them is usually unknown. Given the lack of information, investigation, and support around this topic, the number of studies regarding rare diseases is still scarce, and even less so the ones that use Machine Learning to support them. With this in mind, the present research aims to develop a Machine Learning model capable of assisting medical decisions in this field.

There are many manifestations and symptoms associated with rare diseases, but a common consequence is brain cell degeneration, which in turn causes the deterioration of patients' motor capabilities. The treatment process for these patients can involve several physiotherapy sessions, which imply high costs and drawbacks. For these reasons, it is becoming more and more important to distinguish a respondent patient from a non-respondent. In this way, the principal goal of this research was to develop a Machine Learning model capable of predicting the improvement of the motor functions of patients after attending physiotherapy, determining whether they will respond or not to the treatment. These predictions provide a base for planning the clinical treatment and assisting in medical decisions.

Artificial Intelligence and Machine Learning can be particularly beneficial in rare disease scenarios, given their ability to uncover complex data patterns and memorize data as no human being can. However, the lack of data is the main drawback for both rare disease case studies and Machine Learning models. In this way, Data Augmentation techniques were further explored to generate more data, enhancing the conditions to make use of Machine Learning algorithms. Besides that, this research resorted to several types of predictive models, such as the Regularization models, which are well known for their capability of generalizing new data. Generalization is an essential characteristic when dealing with such unique and different diseases. Besides that, ensembles of models were also considered since Ensembles boost the learning of individual algorithms and avoid overfitting. Ensembles turned out to be the best model to predict the overall patients' motor functions after the therapy, with an error rate of approximately 4%.

KEYWORDS

Machine learning; Rare Diseases; Data Augmentation; Data Mining; Small Data; Imbalanced Regression

INDEX

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

AI	Artificial Intelligence
ML	Machine Learning
SMOTER	Synthetic Minority Oversampling Technique for Regression
SMOGN	SMOTER and Gaussian Noise
LOC	Local Outlier Factor
NN	Neural Networks
SVM	Support Vector Machines
TBI	Traumatic Brain Injury
GMFM	Gross Motor Function Measure

1. INTRODUCTION

With the evolution of Machine Learning and its applications comes the discovery of new ways to solve real-world problems hardly understood by humans. Accordingly, Machine Learning is becoming a crucial tool in healthcare. For instance, it helps simplify administrative procedures in hospitals, assists in the identification of several types of diseases, and personalizes and improves medical treatments.

The current project aims to apply Machine Learning techniques in the medical field, with the principal goal of predicting mobility improvement in patients with rare diseases after attending physiotherapy.

1.1. RARE DISEASES

When a disease affects fewer than 5 people in 10 000 in the European Union, it is defined as a rare disease. It is estimated that between 5 000 and 8 000 rare diseases exist which affect over 30 million European Union citizens [1] and approximately 350 million people globally [2]. Given the low patient population for each individual disease and consequent lack of information, the investigation of this topic is still very scarce. This results in reduced incentives for companies to develop diagnosis and treatments to handle these diseases [2]. Thus, finding treatment can be extremely difficult and a cure near on impossible to identify. Moreover, rarer conditions, such as extreme cases associated exclusively with one patient, require even more personalized medical care.

1.1.1. Causes and Consequences

Rare diseases can stem from several different causes. Genetic conditions are the most common causes (72%), such as unusual changes in genes or chromosomes. They can also be caused by infections (bacterial or viral), allergies, or even by environmental causes [3]. They can be hereditary as well, but the exact cause of many of these rare diseases is still unknown and in a still ongoing discovery process.

The consequences and effects are also usually unknown, without clear insight into what the diseases may cause or influence in the future. Effects may vary drastically depending on the disease, severity, and patient, but they are often chronic and potentially fatal. About 70% of rare genetic diseases begin in childhood [3]. 70% of rare disease patients are children, and around 30% of diagnosed children do not accomplish their fifth birthday [4].

These diseases are usually debilitating, so the life quality of a rare disease patient is affected by the lack or loss of autonomy due to the chronic, progressive, and degenerative aspects [3]. Due to brain cell degeneration, mobility is often affected. In the present research, the effect on the patient's mobility will be the subject of study. In this way, the patient's mobility was measured and evaluated to determine to what extent their condition was damaging their ability to move.

1.1.2. Diagnosis and Treatment

Once again, due to diseases' diversity, lack of basic knowledge, and non-quality information, a correct diagnosis can be extremely difficult to accomplish. A survey revealed that it takes two to three misdiagnoses on average before a patient receives the correct diagnosis, taking over five years to achieve it and needing to consult about eight specialists [5]. In fact, several studies concluded that it is more important to receive the correct diagnosis than the appropriate care [6].

Moreover, there is no known effective cure for the majority of the diseases. More precisely, only 5% have a treatment. Nevertheless, there are medications to treat the symptoms. These are reported as 'orphan drugs' since they attempt to treat unusual diseases rarely sponsored, requiring atypical marketing campaigns and weak commercial incentives [2, 4]. Besides that, there are also therapies and other approaches available to try to relieve the symptoms.

Finding the appropriate treatment becomes even more difficult when the diagnosis is incomplete. Medical doctors and medical experts cannot be aware of every rare disease due to the limited capability of the human brain to memorize and process every information and corresponding details. In fact doctors will only occasionally encounter rare diseases in practice. Even rare disease experts will not have comprehensive knowledge of all uncommon diseases [7].

1.2. The Role of Machine Learning

Innovative approaches are required, given the specific challenges and difficulties faced when dealing with rare diseases. This situation demands automated and robust systems capable of integrating, processing, and memorizing all the information available from several sources. In this way, Machine Learning can be particularly beneficial to use in rare diseases in order to replace burdensome work currently done manually [7, 6].

Machine Learning also has the power to enhance the conditions offered for detecting and treating patients' illnesses. Besides that, it can reduce healthcare costs and errors. On these grounds, some studies in the field of Artificial intelligence have been published to try to help in the fight against rare diseases. Research [5] revealed that in 10 years, only 211 studies were conducted using Machine Learning to investigate approximately 74 rare diseases, within a total of over 5 000. In general, most registered studies were conducted when dealing with a disease with a higher prevalence in the population. The majority of the studies used Machine Learning for diagnosis (40.8%) or prognosis (38.4%). However, studies with the goal of improving treatment were very limited (about 4.7%). In addition, the patient numbers in each research was typically in the low hundreds, ranging from 20 to 99.

As the solutions to such a critical issue are so scarce, addressing this situation would significantly impact public health, considering nearly 4% of the worldwide population is affected, and 3 to 10% of all hospitalizations are related to a rare disease.

1.2.1. Machine Learning in Raríssimas

Considering this situation, several specialized medical centers, such as Raríssimas, have been established in recent times to aid people affected by rare diseases. Raríssimas is a non-profit organization that supports patients and families that live together with rare diseases in order to improve their quality of life and raise awareness in the community. Casa dos Marcos, integrated into Raríssimas, was the first and only infrastructure in Portugal dedicated to assisting and accommodating patients with rare diseases. They provide support through a set of specialized services, which include: clinical unit ambulatory care, integrated continuing care unit, physical and medical development, and rehabilitation (including physiotherapy and hydrotherapy), along with speech and language therapy. Moreover, they provide autonomous residence units and occupational activities. All the facilities are open to the community thereby extending this assistance to society in general [3].

Since Raríssimas is the largest specialized organization in the Iberian Peninsula dedicated to rare diseases, it is of great relevance to assist them in handling this issue. On these grounds, a study proposing the usage of Genetic Programming [8] was recently integrated into an application in the Raríssimas organization to assist medical decisions for treating rare diseases. The present research objective is to use further Machine Learning techniques to reduce the error of Genetic Programming models and, beyond that, try to provide more interpretable results to offer a reliable and effective source for the community to rely on.

1.3. RESEARCH QUESTIONS

As mentioned previously, the principal aim of this project is to construct a model capable of predicting mobility improvement in patients with rare diseases after attending physiotherapy sessions, resorting to data provided by the Raríssimas organization. With this objective in mind, the following research questions can be considered for answering:

Broadly speaking:

A. Is the therapy effective for the patients?
B. Which therapy exercises have the highest impact on the therapy result?

And given a patient's historical data:

C. Is the patient going to respond to the treatment?
D. How much mobility is the patient going to recover?
E. Which functional motions will the patient improve?
F. Is it worth allocating efforts for this patient at the moment?

And finally:

G. Are at least some Machine Learning methods able to learn reliable and robust models even in presence of such small data and different conditions?

1.4. STRUCTURE OF THE THESIS

The present document is organized into six chapters, including the current one, each explaining an integral piece that constitutes this research:

- **Chapter 2** – Theoretical Background – Introduction to Rare Diseases concepts and Machine Learning techniques
- **Chapter 3** – Literature Review – State of the art techniques
- **Chapter 4** – Methodology – Methods used to conduct the research, including a description of the dataset and most appropriate Machine Learning techniques
- **Chapter 5** – Results and Discussion – Methodology quantitative and qualitative results and respective discussion
- **Chapter 6** – Conclusions – Research conclusions and respective limitations and recommended future work

2. THEORETICAL BACKGROUND

In this chapter, the context of the clinical problem will be described, as well as an introduction to the basics of Machine Learning and a further explanation of the techniques required to present the methodology and results of this research.

2.1. RARE DISEASES PATIENTS' MOBILITY

As mentioned in the previous chapter, patients' mobility deterioration may be one of the many clinical manifestations of rare diseases. On these grounds, several therapies have emerged during the last years to enhance patients' mobility and subsequently increase their day to day life conditions. PediaSuit protocol is one example of the many available treatments that have been developed in order to relieve the patient's symptoms and prevent the degradation of their motor capability.

This research is focused on predicting the improvement in the patient's mobility when the PediaSuit physiotherapy is applied since this is the therapy currently being used in the Raríssimas organization.

2.1.1. PediaSuit Protocol

PediaSuit has been created as an intensive physical therapy to assist mostly children with neurological disorders like cerebral palsy, brain injuries, and other conditions that cause patients' motor disabilities. It consists of a customized treatment plan that includes specialized and rigorous exercises that reduce abnormal reflexes and encourage the development of new, correct, and functional motions [8]. The program covers approximately 80 hours, up to 4 hours of therapy a day, 5 days a week, throughout a 3 to 4 week period. Compared with traditional amounts of physiotherapy, this intensive therapy has significantly improved and accelerated results [9]. However, this program requires a lot of resources, including specialized therapists, clinical sites, equipment, and technologies, resulting in a significantly high cost to the average citizen. The cost of this therapy can range between 1300 and 2500 euros, depending on the required number of hours per session [8].

To try to understand to what extent this intensive rehabilitation program results in progress for the patients, their mobility performance is evaluated with the GMFM-88 tool. This evaluation is made before the start of the therapy, and the patients are re-evaluated at the end.

2.1.2. Mobility Measurement

The Gross Motor Function Measure (GMFM) is a standardized assessment tool designed and validated to measure the change in gross motor functions that occur over time in children with cerebral palsy. The GMFM includes 88 test items (exercises) where a patient is evaluated, so each item contains a specific description of an activity. The therapist asks for the patient to perform each activity, and the patient is therefore punctuated based on the following scale:

Scale	Meaning
0	Does not start
1	Starts
2	Completes partially
3	Does not complete
NT	Not tested

Table 1: GMFM evaluation scale

The 88 items are organized into different categories, which are accordingly sorted by degree of difficulty:

Category	Meaning	# Items
A	Lying and Rolling	17
B	Sitting	20
C	Crawling and Kneeling	14
D	Standing	13
E	Walking, Running and Jumping	24

Table 2: GMFM exercises categories

One example of an exercise belonging to category A is "turns the head with symmetrical limbs" [10]. Then, these items are summed and normalized for each category to present a general performance score in each area (summary indicators). A final category G (Global) is calculated by averaging all these presented scores. Category G provides a general overview of the motor capability of the patient to the therapist.

Although GMFM-88 has emerged to measure motor disabilities in children with cerebral palsy, it can be extended to other diseases. For this reason, Raríssimas decided to use this tool to evaluate the improvement of their patients with rare diseases after attending the PediaSuit physiotherapy. In this way, GMFM-88 is used to measure the patient's mobility twice: before and after the therapy.

2.2. MACHINE LEARNING

As Artificial Intelligence becomes more and more crucial in our daily lives, it is providing opportunities for the evolution of Machine Learning in the medical field. Whether helping the medical staff to identify if a patient has cancer or not; trying to predict the probability of one developing Alzheimer's disease; what are the most appropriate medications for a specific illness; Machine Learning models can support all these challenges.

Machine Learning is a subfield of Artificial Intelligence and can be defined as algorithms that learn as data is provided. So, as time passes and data is fed, they are able to uncover complex data patterns [2] and develop "intelligence".

2.2.1. Machine Learning Categories

Predictive Models

Predictive models aim to predict an outcome based on previously given information. To be possible for the algorithm to predict the outcome, many situations where the outcome is known are provided to the model. Thereafter, it will try to find the relationships between the input and the output and consequently "learn". The input information is named independent variables, and the outcome is denominated as the target variable [11]. Predictive algorithms have the power to forecast the future making predictions on new data with the trained algorithms.

Predictive models are subdivided into two categories: Classification problems, where the target variable is categorical (for example, trying to predict if a patient has cancer or not given his radiological images), or Regression problems, where the target variable is numerical (for instance, predict the length of a patient's hospital stay based on clinical registration data).

Descriptive Models

In contrast with predictive models, descriptive models do not try to predict a target variable. Instead, they search for patterns and structures among all the available variables [11]. Descriptive models are commonly used for clustering purposes. For instance, health insurance companies usually try to find alternatives to a "one size fits all" product and use descriptive models to identify clusters of clients with similar behaviors and promote different products to each.

2.2.2. Data Augmentation

Machine Learning algorithms rely on data to discover patterns and acquire knowledge over time. When faced with small datasets and therefore not feeding the algorithms with enough data, the result will have decreased performance. In other words, satisfactory results depend on the amount of data available. Peter Norvig, Google's research director, stated, "We don't have better algorithms. We just have more data."

In an attempt to solve this problem, several techniques of data augmentation and generation have been developed. Data augmentation aims to increase the number of samples in a dataset by creating copies or slightly modified copies of the existing samples or even by generating synthetic data.

2.2.3. Imbalanced Data

Data augmentation is usually associated with the concept of imbalanced datasets, this means datasets whose target variable has an uneven distribution of samples in the classes, having a higher representation in one of the classes (majority class) than in others (minority class), resulting in the model's performance decrease.

This imbalance causes biased machine learning models since most learning systems examine the space of possible models to optimize some criteria, and these criteria are often linked to an average performance. The problem is that the average performance will typically reflect only the most common scenarios. The purpose of imbalanced-aware sampling techniques is to avoid this mistake by concentrating the training on every situation, not only on the most common [12].

2.2.3.1. Over and Under-Sampling

The over-sampling technique attempts to address both problems of lack of data and imbalanced classes by increasing the observations present in the minority class of the dataset. More specific approaches, such as Random Over-sampling, randomly choose observations to duplicate.

The main disadvantage of this resampling technique is overfitting since the same observations from the dataset are fed to the model multiple times. This can result in the inability of the model to generalize to data outside of the training data. In other words, the algorithm cannot perform accurately against unseen data, failing the goal of the ML models [13].

Under-sampling can also balance the dataset by reducing the number of observations in the minority class. However, this can lead to the loss of valuable information for the model.

2.2.3.2. SMOTER

SMOTE for Regression (SMOTER) adapts the SMOTE algorithm [13] for regression problems. SMOTE aims to balance the proportion of samples in the classes of a dataset in a classification problem. This can be achieved by combining over-sampling and under-sampling, more precisely by increasing the number of observations in the minority class and reducing them in the majority classes. This technique not only increases the number of samples in the whole dataset but simultaneously simplifies the algorithm's learning process by feeding it with the same amount of data for each class.

Several methods counterattack the problem of unbalanced datasets for classification problems. However, the quantity of studies shrinks when dealing with regression problems [12]. SMOTER bypasses this situation.

Beforehand, SMOTER applies a relevance function to the target variable, identifying the rare and frequent values within the continuous variable. In this way, infrequent values can be considered of more importance than frequent values. It is important to note that it is also possible for the user to specify a threshold on the relevance values. Two steps follow this. First, SMOTER increases the number of samples of rare target cases considering the relevance function's results. This is accomplished by generating synthetic cases based on the SMOTE technique. The synthetic data is obtained by calculating the k-nearest neighbors of the rare cases. Then by interpolating the feature values of a random neighbor with the rare case, a new observation is created, having the value of the target variable equal to the weighted average of both cases. Secondly, under-sampling is applied to the common values in order to balance the dataset, also considering the results of the relevance function [12].

The fundamental goal of SMOTER was to adapt SMOTE to regression problems, but besides that, this method has the advantage of supporting categorical variables, unlike SMOTE.

2.2.3.3. Gaussian Noise

The Gaussian Noise strategy initially introduced for classification tasks [14] uses normally distributed noise to create new synthetical data. Small random perturbations are added to the replicas of samples from the minority class to treat high-class imbalance and consequently improve models' generalization.

Adapted Gaussian Noise (GN) to regression also combines under-sampling of common samples with over-sampling of the rare samples, as SMOTER. However, as mentioned previously, the GN strategy uses normally distributed noise, whereas SMOTER uses interpolation [15].

2.2.3.4. SMOGN

SMOGN is a further pre-processing approach to handle imbalanced regression. It combines under-sampling with over-sampling, as the previously presented techniques.

To decide which samples to under or over-sample, this method also requires the relevance function results to define the common/rare cases for the regression problem. It is important to note that the user can specify the percentage of under and over-sampling. Random under-sampling is then applied: random observations from the common cases are selected to be eliminated. Afterward, two previously presented strategies, SMOTER and GN, are combined to apply oversampling to rare cases. In this way, two existing solutions are implemented to address issues identified by both of them. SMOTER has the advantage of introducing more diversity to the data generation to counterattack the conservative GN approach. Hence, for each rare case, SMOTER generates artificial data with the k-nearest neighbors (safe area for interpolation). GN is then used with neighbors that are not so close (unsafe regions), where the distance from the rare case is considered too high to apply interpolation [16].

In Figure 1, the synthetic observation created based on a rare case (rare cases marked with points and common with crosses) is represented in red. It is possible to observe the safe and unsafe areas to perform the interpolation. Further details regarding the figure below can be found in [16].

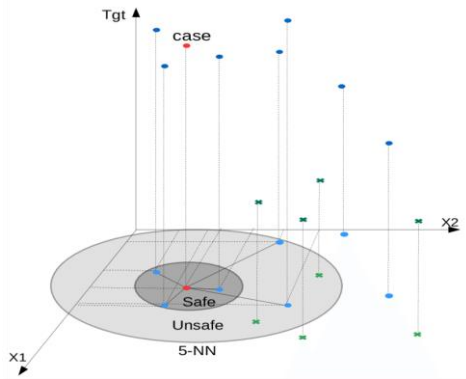

Figure 1: SMOGN Oversampling example

2.2.4. Regularization

One common issue of ML models, especially when dealing with small datasets or having too many dimensions, is overfitting. As mentioned, overfitting happens when the model fits precisely to the training data, resulting in the inability to generalize to new data. Overfitting can be avoided by using regularization techniques, which penalize the model's fit using smoothness criteria [14]. On the one hand, this increases the bias (error in training data), but on the other hand, the variance (error on new data) is going to decrease [11].

Some ML models are more suitable for handling this dilemma than others, such as Linear models, which tend to have higher bias but lower variance.

2.2.4.1. Linear Regression

An ordinary Linear Regression attempts to describe the relationship between independent variables and the target variable using a linear approach that can be written in the following form:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + e_i,$$

where y_i represents the target value for the ith sample, x_{ip} represents the pth independent variable value for the ith sample, and β_p represents the estimated parameters. More particularly, β_0 represents the estimated intercept, and remaining β_p represents the estimated coefficients of the regression model for p independent variables. Finally, e_i represents the residual factors plus possible measurement errors. This model seeks to find the estimates of the coefficients that provide the minimum bias, more precisely, that minimizes the sum of squared errors (SSE) [17]. This method is called Ordinary Least Squares (OLS), and it is described in the following equation:

$$\hat{\beta} = \arg\min \sum_{i=1}^{n} \left| y_i - \sum_{j=1}^{p} X_{ij}\beta_j \right|^2 = \|y - X\beta\|^2$$

The main advantage of Linear Regression is that it is possible to understand the direct impact of each independent variable on the target variable. Just by multiplying the variable's value of a sample by the respective coefficient, it is possible to calculate the influence of that given variable on the outcome, making Linear Regression one of the simplest and most interpretable models [17].

2.2.4.2. Penalized Regression

The principles of linear regression drive penalized Regression models, but instead of finding the parameters that provide the minimum bias, they try to find the estimates that ensure a lower variance. They penalize the fit of the linear model by introducing additional constraints into the optimization function, providing an ideal bias-variance trade-off, and avoiding overfitting.

Ridge, LASSO, and Elastic Net Regression are some examples of penalized regression models, known as regularization models.

Ridge Regression

Ridge Regression, or Tikhonov regularization, adds an extra parameter to the linear regression equation (λ parameter). More precisely, it applies OLS with l2 regularization:

$$\hat{\beta} = \|y - X\beta\|^2 + \lambda\|\beta\|^2$$

By incorporating the regularization penalty (λ), the OLS loss function will increase, resulting in a penalty on the size of the parameter estimates ($\hat{\beta}$), pushing them towards zero. This means that the coefficients will have lower values, meaning that the independent variables' values will not have as much influence on the result of the target variable as they once had [17]. The higher the λ value, the higher the regularization effect (less sensitive are the independent variables to the target), consequently, although the higher the bias, the lower the variance.

Ridge Regression is often used when in the presence of highly correlated independent variables (multicollinearity). This situation is undesirable for most models since "duplicated information" can only harm the model, leading to higher variance [17]. Ridge balances the coefficients of correlated variables, distributing equal weight for the correlated variables, corresponding to the same value if there would only exist one of them [18].

Ridge also has other parameters that deserve further explorations, such as:

- Parameter *tolerance for optimization* – this value is mentioned as the "precision of the solution". It corresponds to the minimum value admitted for the duality gap, the difference between the primal (minimization) problem and the dual (maximization) problem, which is always positive. Although, in the case of strong duality, which happens in LASSO problems, the duality gap can be zero. This way, the optimal solution is closer when the tolerance value is shrunk.
- Parameter *maximum number of iterations* – defines the limit of the number of iterations that the model will perform if it does not converge before.
- Parameter *solver* – there are several options available for this parameter. Each represents a different algorithm that attempts to find the optimal coefficients weights that minimize the loss function. Some examples are the 'sag' (which stands for Stochastic Average Gradient Descent); the 'l-bfgs-b' (adaption of the Limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm); and many others.

LASSO

A popular alternative of Ridge Regression that only shrinks the coefficients asymptotically close to zero is LASSO (Least Absolute Shrinkage and Selection Operator), which can assign the actual zero to the coefficients. In this way, if the coefficients are equal to zero, the model considers the corresponding

variables unimportant. This is possible by using a similar penalty to Ridge Regression, but instead of using the l2 norm, the l1 norm is chosen [17]:

$$\hat{\beta} = \|y - X\beta\|^2 + \lambda\|\beta\|$$

Another way of avoiding overfitting is to reduce the number of features. In other words, to perform feature selection and choose the essential variables that should be kept and which to throw away [19]. LASSO model aims to do this, unlike Ridge regularization, which keeps all the features and only reduces the magnitude of the parameters.

Besides having the regularization penalty (λ), the tolerance for optimization, and the maximum number of iterations, as Ridge Regression, it is also possible to control other parameters in LASSO models, such as:

- Parameter *selection* – defines how feature coefficients are updated, whether individually and in a 'cyclic' way, where all coefficients are updated; or 'random' coefficients can be chosen to update at each iteration.

Elastic Net

Elastic Net has emerged to combine both Ridge and LASSO Regression advantages. This model implements an effective regularization employing the Ridge penalty and has a strong feature selection power from the LASSO penalty [17]:

$$\hat{\beta} = \|y - X\beta\|^2 + \lambda_1\|\beta\|^2 + \lambda_2\|\beta\|$$

As Elastic Net combines both penalties, it is possible to control the amount of l1 and, consequently, l2 penalization by defining the l1 ratio.

2.2.5. Tree-Based Algorithms

Linear models are most suitable when facing linear relationships. Otherwise, other options have to be explored. One of the most used algorithms that also provides a straightforward interpretation is the Decision Tree. The main goal of tree-based algorithms is constructing a tree composed of decision rules, forming simple logical statements such as "if a variable value is greater than X, then the most probable final prediction is Z" [17]. Each logical statement ("if") corresponds to a split in a chosen variable, and these splits are defined by selecting the best possible split in the dataset.

A particular pitfall of decision trees is that they are prone to overfitting since they always try to obtain the best split at each step, only stopping when the error rate is equal to or near zero, resulting in partitions to be as pure as possible. Nonetheless, it is possible to control overfitting. Instead of allowing the tree to grow until it reaches perfection, one could remove parts of the tree, avoiding overfitting. This technique is called pruning and can be managed by adjusting the following parameters:

- Parameter *maximum depth* – controls the maximum number of levels (depth) allowed for the tree to grow.

- Parameter *minimum samples split* – the minimum number of observations required to perform a split.

Many other parameters are available, being these are the ones that have the highest impact on the overfitting.

2.2.6. Ensemble Methods

Ensemble methods aim to increase robustness over a single estimate by combining predictions of several estimators. They work through averaging or voting approaches from the predictions of each estimator, trying to find a "consensus" prediction [20]. In this way, these methods can improve the final results by reducing the bias and/or variance [11], increasing the generalization ability.

2.2.6.1. Bagging

Bagging is a specific type of ensemble that uses a single base estimator. The difference from model to model in the ensemble is that each will be fed with a different dataset that will be obtained by bootstrapping the original dataset, that is to say, by getting a random sample of the original data with replacement [21]. This introduces diversity in each model, resulting in different predictions that will be averaged to provide the final predictions in the final. In this way, it is possible to control the following parameters in a Bagging model:

- Parameter *base estimator* – the model, used as the base learner for the ensemble.
- Parameter *number estimators* – the amount of base learners.
- Parameter *maximum number of samples* – the number of observations that will be bootstrapped from the original dataset to train each base learner.

Random Forest

Random Forests reduce the high likelihood of overfitting that decision trees tend to have by combining the results of several decision trees. Each decision tree is fed with a different dataset, by using bootstrapping, just like in bagging. Besides that, the features feed to the algorithms are also different by randomly choosing a subset from the initial ones. In this way, the parameters that are possible to adjust in the Random Forest are similar to the ones in Decision Trees and Bagging.

2.2.6.2. Stacking

A further ensemble technique is Stacking, also known as Stacked Generalization. This technique uses the output of several base models and then feds them to a final model that makes the final predictions. The advantage of Stacking is that it may use a variety of effective models to accomplish the final task ending up performing better than any of those models. However, it is recommended for this final

model to be simpler than the base estimators, as it is not desired to reduce the strength of each learner.

- Parameter *estimators* – base models that are going to form the stack.
- Parameter *final estimator* – a final model that will combine the predictions of each of the base estimators.

2.2.7. Model Evaluation

In order to determine the best Machine Learning model for a given problem, it is necessary to assess and compare the models. Several evaluation metrics quantify a model's performance, providing an easy way of checking whether the results are satisfying or not. In this way, the metrics for regression problems are presented below.

2.2.7.1. MSE

Before presenting the first evaluation metric, it is essential to clarify what an error is. An error, or residual, is the difference between the predicted value by the model and the actual value. In order to know the overall performance of the model, it is possible to calculate the average of the errors. However, if a standard average is computed, the negative values will cancel the positive values. The Mean Squared Error (MSE) overcomes this problem by squaring the errors. The MSE formula is represented below:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Where y_i represents the actual target value for the ith sample, and $\hat{y}_i$ represents the predicted target value by the model.

2.2.7.2. RMSE

Root-Mean-Squared-Error (RMSE) measures the magnitude of the error, and it is the most common method to quantify the model's predictive capability [17]. RMSE calculates the square root of MSE so that the result will be in the same units as the target variable. Thus, it can be more easily understood.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Since the errors are squared before they are averaged, the RMSE gives a relatively high weight to large errors. On the one hand, this can be useful when large errors are particularly undesirable. On the other hand, in the presence of outliers, RMSE can be jeopardized, resulting in a too high value.

2.2.7.3. MAPE

The Mean Absolute Percentage Error (MAPE) calculates the absolute error and normalizes it by dividing it by the actual value. This is computed for every observation. Then the average is calculated. By multiplying the result by 100, a percentage is obtained. One of the main advantages of MAPE is its independence from the scale of data, as it represents a percentage.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|\hat{y}_i - y_i|}{y_i}$$

2.2.8. Model Interpretability

The most common approach to solving a Machine Learning problem is to gather a set of models, test them all, and choose the one with the lowest error rate. Usually, this process does not consider whether the model is too complex, leading to the use of black box models, which are models that are not interpretable by humans, given their extreme complexity. With this in mind, several techniques were developed to bring interpretability to the models [22].

The principle of Ceteris-Paribus emerged to provide meaning to the models' predictions. The latin expression "Ceteris Paribus" or common “Coeteris Paribus” means "all other things remain unchanged". In Machine Learning, this concept is applied by studying the impact of a variable on the target, assuming that the values of all other variables remain constant. Thereafter it is possible to analyze and understand how variations in the variable's values impact the model's predictions.

3. LITERATURE REVIEW

After a brief explanation to the context of the problem and an introduction to some Machine Learning techniques, this chapter will present an overview of the related work already written on this research topic. This way, a short introduction will be given as to why certain techniques were chosen or excluded from analysis during this research based on knowledge acquired from previous experiences.

3.1. MACHINE LEARNING IN RARE DISEASES

As mentioned previously, in 2020 it was estimated that only around 200 studies using Machine Learning in the rare diseases field existed, and even less (38%) had the final goal of predicting a patient's prognosis [5]. This research aims to contribute to this topic that needs further exploration since it is a significant public health problem.

It is important to note that Artificial Intelligence has already positively impacted society. For instance, in [7] was found that integrated Artificial Intelligence techniques into decision support systems can help accelerate rare disease diagnoses, reducing the excessive amount of time and trouble in this process. Moreover, Machine Learning models, not only provide ways to predict rare cancer diagnosis and prognosis, but also replicate several therapeutic scenarios, delivering better individualized treatments [2]. In [23], even simpler ML algorithms such as K-Nearest Neighbor (KNN) can predict the rehabilitation potential more accurately than standard clinical evaluation protocols or other common practices. AI and Machine Learning are then becoming widely used in the medical domain and are beginning to match, and even exceed, human performance in several areas [5] since the human brain has a limited capacity for process and memory [7, 6].

Furthermore, it is becoming critical to identify a respondent patient (who responds positively to a treatment) and a non-respondent patient. It is crucial to ensure the quality of the treatment provided to avoid unnecessary approaches and guarantee patient improvement [24]. Machine Learning can also support these medical decisions, improving their efficiency, as shown in [24], that with the help of self-learning algorithms, such as Random Forest and Neural Networks, it is possible to increase the efficiency of administrations of the complete GMFM-66 assessment exercises.

Much more possible applications of ML in the Medicine field can be found in [6].

3.2. MACHINE LEARNING WITH SMALL DATASETS

The amount of data plays an essential role in developing Machine Learning models, as "A dumb algorithm with lots and lots of data beats a clever one with modest amounts of it" [16]. If the model does not have enough data, it will not understand the patterns in the data. This is the main pitfall when dealing with rare diseases since most studies barely have enough data to train the model. In [5], it was shown that most research on rare diseases has between 20 and 99 observations to study. Besides that, only a few (12%) used external data to validate the accuracy of their models.

There are several ways to overcome this problem in rare disease studies. The first approach focuses on acquiring more data, going beyond the disease(s) of the study, and collecting data from undiagnosed patients or diagnosed with other conditions. One can also use pre-trained models (transferred learning approach) and fine-tune them to the current use case. And lastly, data augmentation can also be considered to generate more data to better train the models [25]. Besides that, of course, making a wise choice of the used models can also help in reducing the difficulty of the problem, for instance, by using ensembles or regularization models.

3.2.1. Data Augmentation

As stated by Professor Glenn Cohen, "medical artificial intelligence is particularly data-hungry" [25]. Therefore, several data augmentation techniques have been used to satisfy this requirement. However, these techniques are much more explored in big data than in small datasets, mainly because they do not provide such a significant impact when applied to small datasets [26]. Nevertheless, as discussed above, populating the dataset with artificial data can be one solution to the main problem in ML with rare diseases. SMOTE, Gaussian Noise, and ADASYN are some of the techniques tested in [26] for clinical datasets that have improved the performance of the models.

3.2.2. Imbalanced Data

Another implication of having small datasets can be the presence of imbalanced data, which is very common in biomedical datasets [26]. Imbalance awareness in rare diseases case studies has improved the results of the models, as shown in [20], that aim to predict rare diseases and common disease-associated non-coding Variants using Machine Learning and applying SMOTE as the primary oversampling technique. It was concluded that ML models applying imbalance-aware strategies successfully outperformed conventional models. Moreover, results of models that do not consider imbalanced datasets are often biased towards the majority of cases, which has a higher cost in situations like diagnosing rare diseases [27].

Furthermore, it is possible to deduce that there are more appropriate strategies to deal with this situation, depending on the dataset type. For instance, the Gaussian Noise technique has presented better results than SMOTE for some clinical datasets on the Jacqueline Beinecke and Dominik Heider research [26]. This method is especially preferred when down-sampling is not an option, as when the dataset is already relatively small, as is often the case in clinical settings.

Although imbalance datasets are a common problem when dealing with rare diseases, from 61 studies using computerized systems for rare diseases' diagnosis, only two proposed imbalanced aware methods [28].

3.3. Machine Learning Models

The conducted studies investigating rare diseases using Machine Learning found in [5] mainly applied Ensemble models (36%), Support Vector Machines (32%), and Artificial Neural Networks (32%). In [24], Random Forest, Support Vector Machine, and Artificial Neural Network were also the chosen models to improve the efficiency of GMFM assessment administration in children with cerebral palsy. For this case study, SVM provided the best results to estimate the GMFM-66 scores. Besides that, the [29] research to predict the treatment response of children with cerebral palsy, using GMFM scores as one of the outcome measures, has used Random Forests, Support Vector Machine, K-Nearest Neighbors, and Logistic Regression as well.

Genetic Algorithms in the medical field are an evolutionary approach that has also gained popularity in recent years. They have also been used in rare diseases, as is the case of the already mentioned study [30], that applies Genetic Programming to assist medical decisions in Raríssimas by predicting GMFM-88 scores after applying PediaSuit physiotherapy to rare disease patients. This study will be used for comparison to the results of the present research.

In conclusion, several ML models have been developed to boost the diagnosis, treatment, and understanding of rare diseases, some more suitable than others. Yet, the "No Free Launch" theorem states that all machine learning models are equally adequate, and there is not a specific ML model that is universally the most effective in every situation. For this reason, all the models mentioned above were subject of study to confirm the one with the highest predictive power for this problem.

Before applying every mentioned model, it is essential to consider the advantages and disadvantages of each type of model to understand if they are suitable for the problem at hand, which will be better explored below.

3.3.1. Regularization Models

Regularization models have come up as an option to consider since they provide low complexity models, and by regulating the model training, they reduce the overfitting probability. For these reasons, they are appropriate when dealing with small datasets. Thus, rare disease scenarios tend to benefit from them. Regularization models such as LASSO, Ridge, and Elastic Net have been used in the medical field, even if only for variable selection, as in [29]. They were also used to construct a prediction model for diagnosing PCPGs, known as tumors of the adrenal medulla [31], considered rare clinical conditions.

3.3.2. Tree-based Algorithms

Although Decision Trees can handle missing values, are robust to outliers, support categorical data [17], and are one of the most interpretable models [25], they are rarely considered the best performing models in a case study. Due to its extreme simplicity and propensity to overfitting, Decision Trees are frequently withdrawn from the winner's podium.

However, tree-based models have been used frequently, such as Random Forest and Gradient Boosting, which use Decision Trees as their base learner model. Mainly because several advantages remain the same, as is the case of Random Forest that remains robust to outliers, which might be helpful when dealing with small datasets and deleting outliers is not an option. Gradient Boosting is also becoming popular. This algorithm was used to construct *Xrare*, which outperforms experts' pipelines in predicting disease-causing variants in genetic features [2].

3.3.3. Ensemble Methods

Ensembles are widely used, given their ability to reduce bias and/or variance. In this way, they can generalize and avoid overfitting better than most models. Ensemble methods can also boost the results when dealing with small datasets [21] susceptible to overfitting. Besides that, they also outperform single models in imbalanced domains [21]. For instance, *CliniPred* tool [2] has achieved a high performance using an ensemble model that incorporates Random Forest and a Gradient Boosting model. Neural Networks are also prone to overfitting if too complex, but it is possible to prevent this issue using ensembles of Neural Networks [32].

3.3.3.1. Random Forest

Random Forest is a popular ensemble method used in several experiences due to its robustness and ease of interpretation. For these reasons, they are often used in rare diseases, as it is possible to observe in [20], where the developed model, *HyperSMURF*, consists of a hyperensemble of Random Forests using SMOTE. The study has also identified that one of the most significant advantages of Random Forest is the ease with which they can be parallelized.

The new tool *VEST* [2] was also developed using Random Forest. *VEST* identifies likely functional missense mutations that alter protein sequences and can therefore be the cause of many diseases. Additionally, the Random Forest model also provides the best score for the prediction of Gross Motor Function Classification System (GMFCS) in the [33] research. Many other studies using Random Forest in the scope of rare diseases can be found in [2].

3.3.4. Other Models

Artificial Neural Networks and Support Vector Machines are also mentioned as popular models to use in rare disease scenarios [5]. However, in [20], it was concluded that classical learning models, such as Artificial Neural Networks and Support Vector Machines, have difficulties in generalization, as they tend to focus only on the majority classes resulting in a low prediction power for the minority classes. Concluding, not all models are appropriated to deal with small and imbalanced datasets.

3.4. MODELS INTERPRETABILITY

Although Machine Learning and Deep Learning techniques applied in medicine have significantly increased over the past years, some specialists prefer to rely on traditional methods. Some consider that since Machine Learning models are not reliable enough, or they are not interpretable, or they just choose not to trust them [23]. In this way, it is crucial to consider the model interpretability and simplicity and consequently choose simpler models over complex ones. Therefore, it will be possible to provide clinicians with an understandable and meaningful result [25].

Linear and Tree-based models, as discussed in the previous chapter, are considered the most interpretable models [25], whereas Artificial Neural Networks and Support Vector Machines are hardly understood by humans [31]. These are called black box models, as they contain highly complex prediction equations [17], leading to misunderstandings. Likewise, complex and extensive Genetic Programming models are also difficult to interpret, which can be seen as the main disadvantage of the final model developed in [8], which is only partially interpretable.

Interpretable models are helpful when, ideally, their explanation is consistent and reliable within the domain they address [25]. Whenever it is impossible to provide completely explainable models, one can also consider using Shapley Additive Values, Ceteris-Paribus, or other methods that quantify the effect of each variable in the models, facilitating the interpretation of its predictions. In [22], it is possible to observe the use of Ceteris-Paribus and other explainable Artificial Intelligence techniques to understand the influence of the variables used in a Random Forest model to predict survival in patients with sepsis infection.

Another procedure that could be used to gain medical specialists' trust is to test the algorithms against human expertise. In [5], from all the collected studies that used Machine Learning to analyze rare diseases, only 3% of them validated the models with medical experts.

4. METHODOLOGY

Previously, all the considered techniques and models were described, and their use was justified. In the present chapter, the methodology approached in this research will be presented. Since this project required all the steps involved in a Data Science project, the chosen methodology was the popular Cross Industry Standard Process for Data Mining (CRISP-DM) [11]. It was considered the most appropriate knowledge discovery process flow, given the need for an iterative and flexible plan. Besides that, all its steps coincide with the precise steps needed to achieve the final objective of the study. The methodology consists of six sequential phases:

- Business Understanding
- Data Access, Exploration & Understanding
- Data Preparation
- Modeling
- Assessment
- Deployment & Monitoring

The first and foremost step of this methodology, included in the Business Understanding phase, is to define the project's final objective. As discussed, this research aims to predict the mobility improvement of patients with rare diseases after applying PediaSuit physiotherapy. This goal will be possible to accomplish by resorting to data regarding the clinical conditions and demographic characteristics of patients with rare diseases. After having the goal identified, then follows the remaining 6 phases, which will be described in the following sections.

4.1. DATA ACCESS

Casa dos Marcos has been recording information about the results of the PediaSuit therapy in some of their patients, wherefore the organization provided a file containing 41 observations with data relative to the patients and their therapy results. This dataset includes patients' socio-demographic data such as age, gender, diagnosed disease, and the number of therapies attended. Besides that, it also comprises the medical history, including the GMFM-88 scores for two different times, before and after PediaSuit treatment, and their corresponding summary indicators.

By taking a closer look at the data, it was possible to understand that there are repeated patients, corresponding to people who have had to go through therapy more than once, concluding that the dataset only concerns 26 different patients.

It is essential to mention that the data available was considered secondary data, so it was not collected for the final purpose of this research. Therefore, not all the patients were diagnosed with a confirmed rare disease (12%). However, in this Data Access phase, case selection is also an essential step since some situations can benefit from other patients' data. For instance, the dataset could include patients that were not yet diagnosed with a rare disease or those with diseases that reveal the same or similar effects of rare diseases, as is the case.

4.1.1. Data Protection

Data anonymity has been a growing concern when dealing with medical data, and GDPR compliance has to be considered for every project using sensitive data. Thus, it is important to note that the process of safeguarding confidential data regarding the patients was guaranteed in this research, ensuring data privacy, protection, and anonymity by not revealing any personal information during its realization.

4.2. Data Understanding and Exploration

After accessing the provided dataset, it is necessary to analyze the data in this initial phase to understand better which methods and algorithms are more appropriate for the following steps. Furthermore, evaluating and ensuring the data quality for the methods used to present the best possible results is crucial. Accordingly, univariate and multivariate analyses will be presented below.

4.2.1. Univariate Analysis

First and foremost, it is necessary to analyze each variable individually to get more detailed information to draw valuable insights and detect possible errors or problems. The following descriptive statistics were drawn from 41 individuals and 108 variables.

- *Age* – Registered patients are, on average, 9 years old, being the minimum 3, and maximum 20. It is possible to observe the *Age* distribution below in Figure 2. The latter value is a possible outlier since this value is detached from the remaining distribution, and in fact, this patient is no longer considered a child.

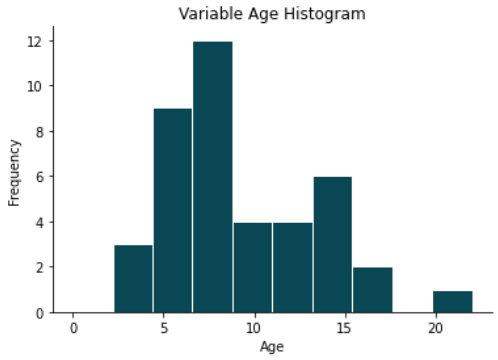

Figure 2: Patients' Age

- *Gender* – The dataset has approximately the same number of female and male children (56% and 43%, respectively).
- *Diagnosis* – There were reported 21 different diseases. From them, 5 samples did not have a diagnosed disease.

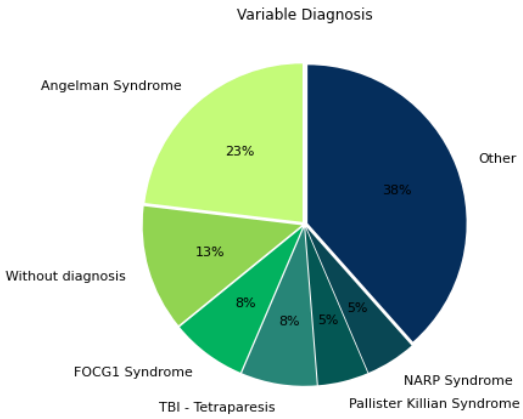

Figure 3: Registered diseases of patients

- *Scores* – The GMFM exercises scores registered before the therapy of sections A (mean value of 88) and B (mean = 80) reported higher values than the ones from Section D (mean = 55) and E (mean = 42). This difference is due to the fact that the exercises in the first sections are more accessible than the remaining ones. In Figure 4, it is possible to visualize the distribution of variables and respective skewness, which suggests that this problem would need more robust models.

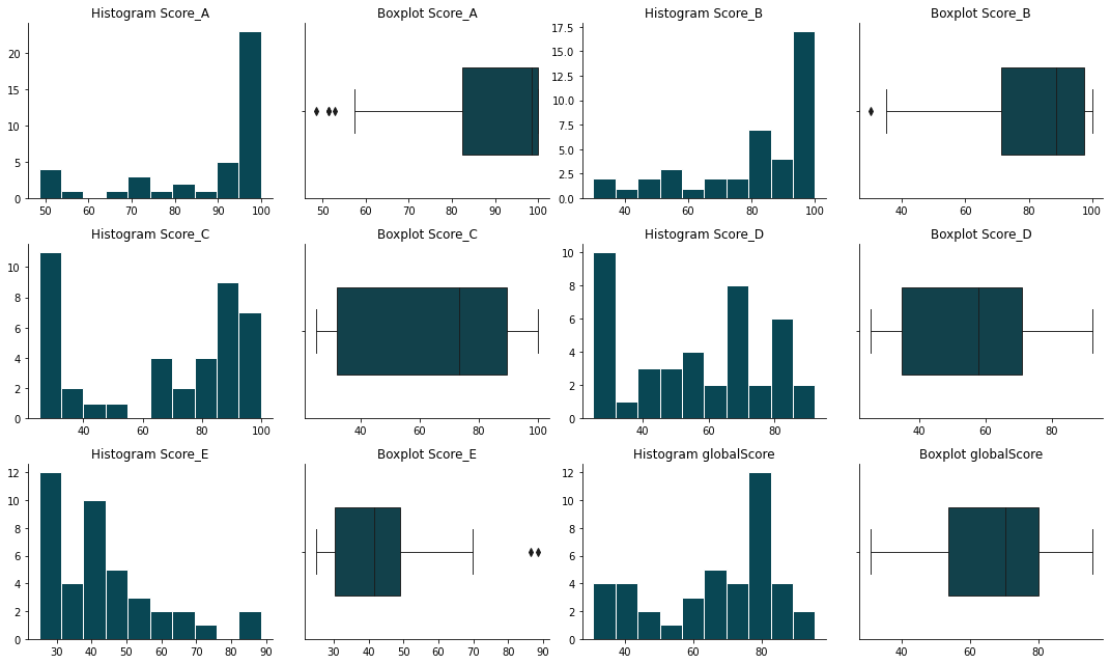

Figure 4: Distributions of Score variables before therapy

- *Scores after* – The GMFM exercise scores registered after the therapy are the target variables. This means that the final models were constructed to predict these values. The target variables included the average score from section A, B, C, D and E after the therapy and the global Score as well. In general terms, scores after the therapy are higher than those registered before the therapy. The average of exercises scores from Section A (mean = 90) and Section B (mean = 86) still have higher values than the ones from Section C (mean = 62) and Section D (mean = 47).

4.2.2. Multivariate Analysis

Multivariate analysis was followed to better understand the relationship between the variables, mainly with the target variables.

- The GMFM results between male and female patients does not seem to vary significantly, as seen in Figure 5.

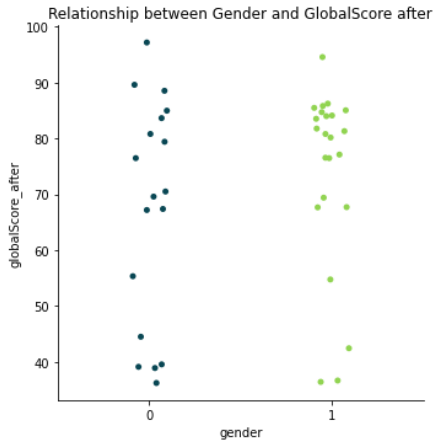

Figure 5: Relationship between Gender and Global score after the therapy

- Regarding the Pearson correlation matrix in Figure 6, which measures the linear dependency between two variables, it is possible to observe a strong positive correlation between most of the score variables, including the GMFM scores, before and after the therapy.

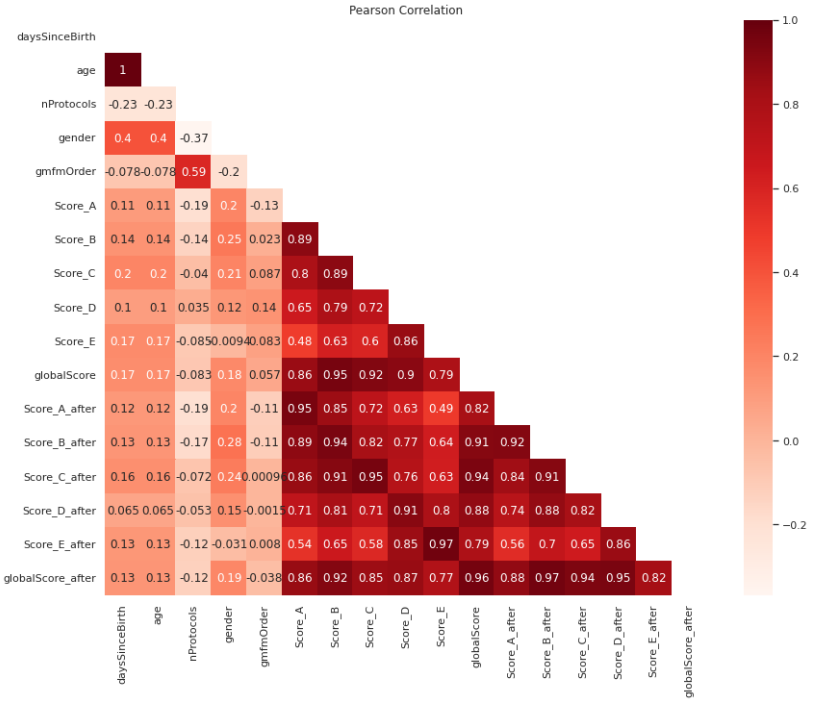

Figure 6: Pearson Correlation matrix

- In the following image, it is possible to validate, in a visual way, the linear relationship between the target variables (GMFM scores after the therapy). Besides that, it is also possible to visualize the skewness of the target variables.

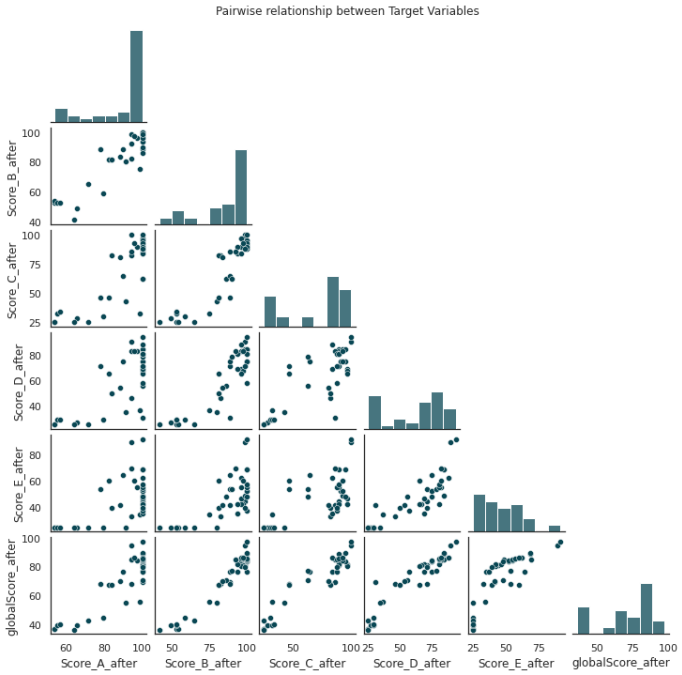

Figure 7: Pairwise relationship between the GMFM Scores after the therapy

4.3. DATA PREPARATION

After obtaining the main insights about the data, it follows the Data Preparation or Pre-processing. This intensive phase covers all the needed steps for transforming raw and noisy data into clean data. This phase is one of the most important since the data will then be used to feed the model, and if the model receives errors and problems, it will return more errors and problems, with unreliable and poor results. Thus, the final objective is to minimize the GIGO (Garbage In Garbage Out) [11].

The main tasks of this phase include: 1) check for any incoherencies or errors and correct them; 2) identify and treat possible outliers; 3) identify and treat the missing values; 4) perform transformations or create new variables from the existing ones.

The provided dataset was in reasonable condition. It did not present any duplicated information; it did not reveal any type of incoherencies, such as negative values or values outside of the expected range; it did not suggest any type of missing values, nor any possible numbers associated with missing values, such as 999. Considering all this, it was clear that the present dataset had the quality to advance to the next steps. However, it was possible to observe the presence of some outliers, which will be better described in the next section.

4.3.1. Outliers

Outliers are atypical, extreme, or inconsistent values and some ML models do not operate smoothly when in their presence [11]. Thus, first of all, they must be identified, which can be done by using visualization tools, such as boxplots or histograms. It is also possible to validate the visual analysis with quantitative methods such as the Z-score or the Interquartile Range [11], or even with anomaly detection algorithms, as the Local Outlier Factor (LOC). LOC calculates the anomaly score of a given sample based on the local density deviation from the surrounding neighborhood. Clustering algorithms, like K-Means, can also be used to identify outliers. By setting the number of clusters to a large number, it is then possible to spot the outliers in the groups that only hold one or two observations, corresponding to clusters isolated from the rest of the data.

After their identification, there are several solutions to mitigate the outliers' effects, although when handling small datasets, some of those solutions, such as simply removing them, are not possible. For this reason, other options had to be considered. Binning of the variables can be done to aggregate the outliers with other values and, in this way, minimize their impact. There are several types of binning. One could consider the width of the intervals, the number of observations in each bin, or the similarity between the values in each bin (using K-Means). Clipping (or Winsorizing) the outliers is also a viable solution. This technique replaces the extreme values by the percentile value that is considered acceptable.

For some variables, clipping presented the best results, yet for others, binning was preferred. In this way, each variable was treated individually with the method that provided the best results to achieve the best possible predictions.

4.3.2. Skewness

Outliers might result from a skewed distribution when dealing with small datasets [17], as it was possible to observe in variable Score A, where most of the values are concentrated on the right side of the distribution, and only a few are visible in the first values. Outliers can be harmful to most of the models, especially to regressions-based models, and the tail of skewed distributions acts like outliers. Along these lines, transformations like the logarithm, square root, or power may be applied to obtain more symmetric distributions, reducing the impact of skewed variables [17].

Alternatively, available statistical methods, such as the Box-Cox and the Yeo-Johnson transformers, automatically perform a family of transformations. By using the maximum likelihood estimation, they determine the best transformation to apply to each variable, whether the log, square, square root, inverse, or others. The main difference between the Box-Cox and the Yeo-Johnson method is that Box-Cox demands the data to be strictly positive, whereas Yeo-Johnson accepts the data in any range.

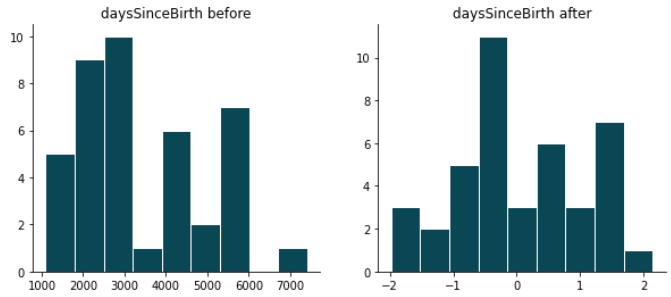

Figure 8: Variable *daysSinceBirth* before and after Yeo-Johnson transformation

In Figure 8 it is possible to visualize the effect of the Yeo-Johnson method in the skewness of the variable *daysSinceBirth*. Before the transformation, the variable registered a moderate skewness (0.58) and after an approximately symmetric distribution (0.03).

Quantile transformer is another option to solve non-standard distributions. This method maps a variable's probability distribution to a uniform or even to a normal distribution through the estimation of the cumulative distribution function.

4.3.3. Feature Engineering

In an attempt to provide the model with further and better information, one may create new variables from the existing ones to obtain new dimensions from raw data and consequently improve the model's performance. This process is known as Feature Engineering.

For instance, from the variable Diagnosis, which identifies the patient disease, it is possible to create a new one that labels whether the patient has an identified disease or if it was not diagnosed yet. Besides that, patients diagnosed with "TBI - Tetraparesis" could be considered to join with patients with "TBI". Moreover, a new variable was created, *increasingFactor*, the resultant of the division between the global Score variable and the number of therapies attended until the moment. This variable revealed to be an essential variable for the models, which will be later discussed.

4.3.4. Feature Encoding

As most Machine Learning models do not accept values in text format, categorical variables must be encoded into numerical ones. There are several possibilities to make this transformation, but the most common approach is to use One Hot encoding. This technique creates dummy variables [17]. In other words, it will create one variable for each category in the categorical variable, indicating whether an observation belongs or not to that category. Thus, for instance, each disease in the variable Diagnosis will have a corresponding variable that will indicate if a patient has that disease (labeled with 1) or not (labeled with 0).

Despite solving the issue related to the requirement of feeding the model only with numeric values, this technique can lead to the curse of dimensionality, especially on small datasets, when the number of dimensions becomes higher than the number of samples. For this reason, this option was excluded since 21 different diseases would create too many variables. However, instead of just excluding this variable from the analysis, diseases that were present in more than 10% of the dataset would be

encoded using dummy variables. As the remaining ones do not have a high enough representation, the variance of the created dummy variable will be almost null, so they can be excluded from the analysis.

Target encoder was also considered since it transforms a categorical variable into a unique numerical variable. On the other hand, as this method uses the expected values for the target in each category to do the encoding, it leads to extreme overfitting. In this way, this option was discarded from the analysis.

4.3.5. Scaling and Standardization

Each Machine Learning model has its characteristics, and consequently, "the need for data pre-processing is determined by the type of model being used" [17]. Some models do not require the data to be scaled (as Decision Trees), whereas all the variables must be on the same scale for models that, for instance, calculate distances. These algorithms give more or less importance to certain variables based on their values' differences. Min-Max normalization [11] was considered for these latter models, as some variables like daysSinceBirth (values between 1000-8000) had a wider range compared with the mobility scores (values between 1-4). Upon application of this method, all values became scaled between 0 and 1.

Besides that, standardization is also one of the most common requirements of Machine Learning models. This means they demand that the variables have a mean equal to 0 and a standard deviation equal to 1. This can be achieved by subtracting the mean and dividing by the standard deviation. Although when in the presence of outliers, the mean and standard deviation is negatively affected and should consequently be replaced by the median and interquartile range, respectively. This is done by using Robust scaler and it is recommended if the treatment of the outliers was not possible to achieve before.

Whenever not required, these methods were avoided since they result in the loss of interpretability of the model as the variables will lose their original values, which is an essential requirement in the medical field. Although it is possible to revert the scaling after the model is trained.

4.3.6. Feature Selection

Feature selection methods have emerged to simplify the models' learning by reducing the amount of available data which models have to ingest and process. This can be accomplished by removing irrelevant and redundant variables [19], avoiding the course of dimensionality problem, which is of great concern, particularly when having small datasets, as discussed before.

First and foremost, it was necessary to verify that the dataset did not present any zero-variance variables, this means, variables that have the same value for all the observations. Zero or low-variance variables lead to the increase of model complexity without increasing its predictive power. In this way, some variables were removed from the dataset. For instance, variable A1 was excluded since only one patient had the value 3, and all remaining had the value 4.

Then, the feature selection methods were applied. These methods can be divided into 3 categories: Filter, Wrapper, and Embedded methods [19]. The filter methods only consider the relationship between a variable and the target variable, while the wrapper considers the model's performance with the selected variables, and the embedded automatically select the variables during the training process.

The filter methods applied in this research were the ones used for regression problems, such as Pearson and Spearman correlation [11], which measure the amount of correlation between two numerical variables. Thus the higher the correlation with the target variable, the better. Irrelevant variables can be discarded if no correlation is found with the target. Additionally, by analyzing the correlation values, it is also possible to understand if there are any redundant variables, which can be identified by checking if there are any independent variables highly correlated with each other [11]. This situation should be avoided since duplicated information will only decrease the efficiency of the model learning [17]. Having this in mind, one of the variables, either daysSinceBirth or Age, had to be discarded to avoid this problem since they are perfectly correlated (multicollinearity problem). Besides that, F-statistics and associated p-values were also evaluated. The F-test analyzes the linear relationship between variables, and the corresponding p-values indicate whether the relationships are statistically significant or not. It was possible to visualize the linear relation between variables (Figure 6 and Figure 7) in the Data Understanding and Exploration section, so with this method, it is possible to quantify it. Furthermore, Mutual Information coefficients were also taken into consideration to quantify the relationships between the variables [17, 19]. Mutual Information statistic is used to calculate the entropy reduction when one variable takes into account the value of another variable, measuring the level of dependency between two variables, so the higher this statistic, the better.

Wrapper methods make use of the model performance to select the best subset of variables, in this way they are computationally more expensive than the filter methods [19]. These methods are subdivided into three categories: forward, backward and stepwise selection [17]. For this research, a backward selection technique was used, more precisely, the Recursive Feature Elimination (RFE) method, that by considering the feature weights, it recursively eliminates the least important variables until it gets the subset of variables that provide the best results. The feature weights are calculated by the model chosen to feed the RFE. It is possible to use, for instance, the coefficients of a Linear Regression to define feature weights. Yet, it is important to note that the variables must be scaled to consider any linear model's coefficients. Feature importance calculated by Tree-based models can also be used to feed RFE.

Last but not least, the Embedded methods, which are models that automatically perform feature selection by themselves. As discussed before, LASSO [19], Ridge, Elastic Net, and Decision Trees have already incorporated a system for choosing the best variables to use in the model. Consequently, after training these models, it is possible to obtain a rank for the features' importance that each method has established.

Lastly, a final rank selector was constructed to combine the results from all the different feature selection techniques mentioned before to form a hybrid feature selection method. Considering all the methods, an average ranking score was calculated for each variable. Then, only the most relevant ones were kept, in order to feed the model only with the strictly necessary information. The main goal of these methods, as shown in [19], is to improve the learning efficiency of models, simplifying the

process of discovering patterns in the data, consequently improving results and reducing the computation time required.

4.3.7. Data Augmentation

A high number of dimensions can cause several issues in a model's performance, but a lack of observations can lead to far more severe problems. As mentioned in the previous chapters, when discussing rare disease cases, one comes across the challenge of scarcity of data. Thus, data augmentation was considered to apply. In this way, SMOTER and SMOGN were the chosen techniques to use. However, the preferred methods were adapted since both perform under-sampling on common target values. Under these conditions, it was decided not to apply under-sampling as the dataset already has a limited amount of data, and important information could be lost if applied under-sampling [26].

The remaining SMOGN parameters were also adapted and finetuned to fit the current dataset better. First and foremost, the relevance of target values must be identified. There are two ways to define them. The first and by default approach is to automatically determine the rare values to be over-sample by the box plot extremes, the values that surpass the whiskers are oversampled. The oversampling also depends on the distribution focus (whether high, low or both). The rare values can also be defined manually when the target variable has a more complex distribution. Besides that, the default number of neighbors considered to create the artificial data was also adapted to a smaller number to fit the present dataset better. The amount of perturbation to apply to the Gaussian Noise also had to be adjusted since too much noise was being introduced to the dataset, resulting in an excessively high variance. However, all the discussed parameters final values depended on which returned the best results.

SMOGN and SMOTER have similar parameters, as both methods rely on SMOTE algorithm. SMOTER has as main parameters the definition of rare and common values for the target, just like SMOGN, and the threshold of rarity, this means until which threshold should an amount of data be considered rare. This threshold was reduced given the nature of the dataset, which is small.

4.3.8. Data Imbalance

As discussed in the previous chapters, a consequence of lack of data is having an imbalanced dataset, meaning that the target has an unbalanced distribution. Having a target variable with outliers, with a skewed distribution, or not even near a uniform distribution can harm models' performance, especially with linear models, since they assume that the residuals (errors) have constant variance (homoscedasticity).

There are several approaches to deal with this situation, which can be within the following categories, depending on where the intervention is made: 1) data-level pre-processing approaches; 2) algorithm-level approaches; 3) hybrid approaches [16, 13]. The first data-level pre-processing approach attempted was already implemented in the data augmentation techniques SMOTER and SMOGN, which have mechanisms that consider unbalanced data.

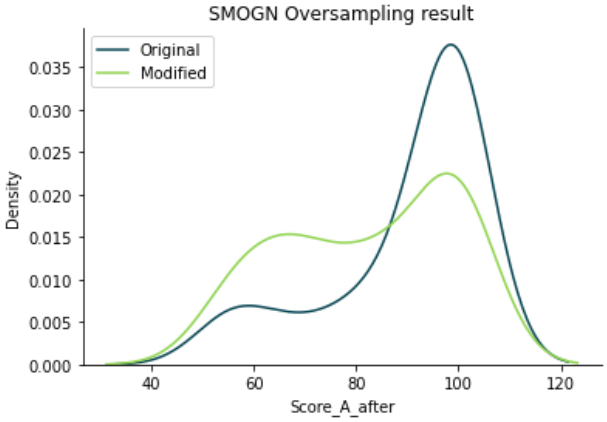

Figure 9: SMOGN transformation effect on the target variable Score A

Another solution to satisfy the homoscedasticity assumption is to reshape the target distribution in the pre-processing phase. This can facilitate the model learning process, allowing for better predictions. Target Transformer was consequently used to apply a given function to the target variable, such as the power or logarithm, stabilizing the variance of residuals.

However, for situations where the problem remains, other alternatives must be found. Algorithm-level approaches are another solution: models or respective evaluation methods adapted to deal with imbalanced datasets. These will be better described in the Modelling section.

4.4. MODELLING

After passing through all the steps of Data Preprocessing to ensure data quality, it is now possible to build and feed the models with the treated data. The questions and goals defined in the Business Understanding phase can now be answered, which is why “the heart of any data mining project is the modeling phase” [11].

Given the nature of the dataset, regularization models, tree-based models, and ensembles were the chosen models since they are the most appropriate for small datasets, as explained in the Literature Review chapter. All these models will be further detailed in the following sections.

4.4.1. Regularization

Regularization models were preferred over the remaining ones since their main goal is to ensure the bias-variance trade-off and, consequently, avoid overfitting, which is a common consequence when working with small datasets. Although in most Machine Learning studies, the bias-variance trade-off is already a requirement, in this specific situation, it is more crucial to achieve a low variance than a low bias. In this way, these models were chosen as they provide a simple way to control the regularization strength, increasing or decreasing the generalization.

Additionally, most of the exercises' score variables present a high correlation between them, and LASSO and Ridge can efficiently manage this situation, unlike a standard Linear Regression model.

Moreover, a significant advantage of any model that is ruled by the equation of a Linear Regression is that it becomes highly interpretable, which is desired for any research in the medical field [17].

Several parameters can be adjusted for each model, which description can be found in the Theoretical Background chapter. The parameter values are briefly discussed below:

Ridge

Besides adopting the penalization parameter, the Ridge algorithm also has other parameters that need to be taken into consideration:

- Parameter λ – the regularization is possible to achieve is by changing this penalization parameter, available in all regularization models: Ridge, LASSO, and Elastic Net. The higher the λ, the higher the regularization effect. As this situation demands models that are able to generalize, this parameter was increased.
- Parameter *tol* – tolerance for the optimization – if too high, the algorithm will stop before finding the optimal solution.
- Parameter *max_iter* – maximum number of iterations – if too low, the model will not be able to converge.
- Parameter *solver* – the default value 'auto' tries to find the most appropriate solver automatically from the type of data. There are other options available, which include: l-bfgs-b, which usually performs better for small datasets compared with others; svd, which uses Singular Value Decomposition of the data, that consequently simplifies the data and removes noise; cholesky, which is faster than svd when the number of samples is less than the number of variables; sparse_cg which is more appropriate than cholesky for large datasets, which is not the case, so it can be excluded from the analysis; and many others.

LASSO

The main LASSO parameters are discussed below:

- Parameter *λ*, *tol*, and *max_iter* – described in Ridge section.
- Parameter *selection* – the random option to select which coefficients to update leads to a faster convergence than cyclic, which can be beneficial when dealing with a large number of dimensions, also because LASSO registers overfitting issues when dealing with high dimensional data [19].

Elastic Net

Elastic Net includes the following parameters:

- Parameter *λ*, *tol*, *max_iter* and *selection* – are described in the LASSO section.

- Parameter *l1 ratio* – if this value is equal to 0, the model will only apply the Ridge penalty. If equal to 1, then the LASSO penalty. It was decided to keep the default value for this parameter (0.5), as a balance between Ridge and LASSO was preferred.

4.4.2. Tree-based models

The Decision Tree is the simplest tree-based model to handle non-linear problems effectively, and it has many advantages, as referred to in the previous chapters. However, it cannot deal with correlated variables [11], whereas regularization models are. Moreover, it is highly prone to overfitting, but there are several ways to control this issue, such as applying pruning techniques or creating ensembles. The latter approach led to Random Forest, which quickly exceeds the Decision Trees results [17]. In this way, this model is explored in the next section.

4.4.3. Ensemble Methods

Ensembles are a solution to control overfitting and model instability [17], not only in Decision Trees but in any model. Ensemble bases its knowledge on several models instead of relying on one. In this way, its performance often surpasses other models' scores.

Bagging

As discussed in the Theoretical Background chapter, Bagging is one type of Ensemble method, and several parameters can be adjusted, which are briefly discussed below:

- Parameter *estimator* – base estimator – a weak model that needs to be reinforced with bagging technique, as Decision Trees or Elastic Net.
- Parameter *n_estimators* – number estimators – the default value is equal to 100. However, when dealing with small datasets, this value should be decreased.
- Parameter *max_samples* – maximum number of samples – the default value trains every base learner with the same number of samples as the original dataset. This number was kept since the number of samples was already too limited.
- Parameter *bootstrap* – set to True since with False, the dataset feed to the base learner would always be the same, leading to no diversity among the models.

Random Forest

Random Forest is similar to a Bagging of Decision Trees. In this way, the parameters in bagging are also available in Random Forest. However, some parameters need further exploration to control the pruning of Decision Trees in the Forest:

- Parameter *n_estimators, max_samples, bootstrap* – described in Bagging section.

- Parameter *max_depth* – maximum depth – by default, the maximum depth is not established, so Decision Trees are allowed to grow until they reach the purest nodes. This leads to overfitting. Thus this value had to be decreased, and the fact that the dataset is small also demands a cut in the depth of the Decision Tree.
- Parameter *min_samples_split* – minimum samples split - within the same line of thought as the max_depth parameter. This number had to be increased to avoid overfitting since its default value equals 2.
- Parameter *criterion* – split criterion – there are several options available, but the most common is the squared error and absolute error. When having an imbalanced dataset, the squared error is the most appropriate option since it penalizes more significant errors, and it is desirable that the model generalizes to every situation.

Stacking

The main parameters in Staking that need to be explored depend on the models introduced in the "stack".

- Parameter *estimators* – a set of models with the highest predictive power for the target variable should be introduced into the stack.
- Parameter *final_estimator* – the final model that combines the outputs from the base estimators should be one of the simplest Machine Learning models available, such as linear models such as Linear Regression or even a Decision Tree. By all means, for linear problems, the preferred model should be Linear Regression.

4.4.4. Multi-Target Regression

Instead of predicting one target variable at a time, using completely different models, and assuming that the targets are independent, one can use Multi-Target Regression (MTR), also known as Chained Models. An MTR model aims to predict multiple target variables instead of just one, and they have emerged to surpass the limitation of combined models with a single output. MTR models are faster to train, and they consider the relationship between the target variables instead of just optimizing the result for each target individually. In this way, MTR Models predict the first target variable that will be fed to the second model, which predicts the second target variable, and so on. Since the current research presents 6 target variables (Score A, B, C, D, E, and global after the therapy) and they are dependent among them, as was observed in Figure 7, it can be beneficial to use MTR models.

4.4.5. Other models

For the remaining attempted models, since their performance was not high enough to justify further exploration, only the results will be presented in the next chapter. Additional details about these models are also discussed in the next chapter.

4.5. ASSESSMENT

Before using the models, it is necessary to define evaluation methods and metrics to evaluate each model individually. Thereafter, it will be possible to determine which is the best-performing model for the current case study.

4.5.1. Evaluation Methods

There are several Model Selection and Evaluation techniques available. Still, it is crucial to remember that all of them provide a way to subdivide the original dataset into a train and a test dataset, ideally with a validation dataset. However, when facing situations where the dataset does not have enough data to be subdivided into 3 datasets, as happens in the present case, the validation dataset must be excluded. All these subsets are created because the model should not be evaluated on the same dataset that was used to build the model, as this would lead to biased results [17].

The training dataset will be used to feed the model, and consequently, the model performance will be tested against the test dataset. It is desired a model that is able to generalize to any situation. Hence the goal is that the model is as effective with the training dataset as it is with the test one. Here it is possible to confirm whether the model is overfitting.

The schemas available for Model Selection are:

- *Hold-Out Method* – divides the original dataset randomly into two parts, train, and test. This method is mostly used when the dataset is large enough to consider that only one partition is enough to rely on the significance and precision of the results [17]. Since the current dataset does not fulfill this requirement, this option was disregarded.
- *K-Fold Cross-Validation Method* – splits the data into k subsets (folds) of equal size. K-1 folds are used as training sets, and the remaining fold is used for testing the model. This procedure is performed k times, and the test fold rotates in each iteration. Then the scores of all k iterations are averaged [13].
- *Leave-One-Out Cross Validation* – is a specific case of K-Fold Cross-Validation, where the k is equal to the total number of observations present in the dataset. Once again, k-1 folds are used for training, and the left-out observation will be used for testing. This is repeated for k iterations. The difference is that this method tests the model against all observations in the dataset. The average performance is then calculated, just like in the previous method. This method is clearly computationally expensive, yet it is the most appropriate option for small datasets.

As the current dataset only presents 41 observations, it was concluded that the best option would be to use the Leave-One-Out Cross Validation method.

In [13], it is clear that when dealing with imbalanced datasets and using Cross-Validation to evaluate the models, one needs to pay attention to over-optimistic results and overfitting. In order to avoid over-optimistic and biased results, for instance, data augmentation must be only applied to the

training dataset for each iteration instead of applying data augmentation to the whole dataset before using Cross-Validation to evaluate the model that would be tested against synthetic data.

Overoptimistic results are not only related to the wrong application of data augmentation techniques. Any other pre-processing steps, such as feature selection, should not be applied before Cross-Validation. This is recommended to avoid data leakage, which means that the model is fed with information outside of the training data, consequently leading to over-optimistic results. For these reasons, pre-processing was implemented within the Leave-One-Out Cross Validation loop. In each iteration, the pre-processing was applied to training data only after the necessary preparation was applied to the test data.

4.5.2. Evaluation Metrics

After choosing the most appropriate technique to evaluate the models, it is necessary to define the metrics that quantify the models' performance. Several metrics were already introduced in the Theoretical Background chapter, and as mentioned before, RMSE is the most common metric and also the most suitable for this problem as it provides an idea of how far the predictions were from the actual value, and at the same time penalizing the highest errors. In this way, the RMSE was calculated across 41 runs for each target variable for both the training and test set. Then an average and the respective standard deviation were calculated.

Additionally, MAPE was also calculated to provide further insights about the models' performance, as it is represented as a percentage and, for this reason, is scale-independent. These metrics will be presented in the Results chapter.

4.5.3. Parameter Optimization

Having all these procedures in mind, it is necessary to define which are the most effective for predicting each target variable. It is important to note that there are parameters that cannot be directly inferred from the data [17]. In this way, the hyperparameters of the estimators and pre-processing techniques were optimized using Cross-Validated Randomized Search and Grid Search [34].

The Randomized Search starts by sampling random values from a pre-defined range for every parameter. Then it searches for the best combination by trying all of them. The ranges' values, as well as the number of desired iterations, are defined by the user. On the other hand, Grid Search attempts all possible combinations between the set of values given for each parameter. Then it returns the best one.

First, an experiment on the Randomized Search was done to understand better where the best values could be situated. After that, it was executed a Grid Search with more specific values, which returned the best parameters. Considering that none of the methods considers the overfitting/underfitting trade-off, a final adjustment to the parameters was made, which final results are presented in the next chapter.

4.6. TOOLS

This project demanded software and hardware capable of parallel processing several experimental sets of models and processing techniques. The software was developed using Python, one of the most widely used programming languages for scientific computing [34]. It is also on the top of the tools used for Data Science. The code was developed in the Google Colaboratory platform, which allows the use of Python and provides a cloud service supporting GPU. Another advantage is the possibility of using the processing power of an Nvidia Tesla K80 GPU.

The Python scripts used mostly libraries from scikit-learn, which provides medium-scale techniques for descriptive and predictive problems [34]. Although scikit-learn is one of the most valuable tools to use for Machine Learning, scikit-learn packages were replaced by cuML and cuDF packages whenever possible. The latter libraries are programmed to use the GPU accelerator.

5. RESULTS AND DISCUSSION

This chapter will present the comparison between the results of all the discussed models and techniques. Besides that, the best results will then be compared with the experiment introduced in [8] that attempts to solve the same problem by resorting to Genetic Programming.

5.1. MODEL COMPARISON

First and foremost, it was necessary to evaluate all the models using the same methods so that one could directly compare them. Then, it was possible to understand which models provided the highest scores for the current dataset and consequently were worthy of further exploration. Below it is possible to observe Table 3, which summarizes the prediction capability of the developed models. The presented results are relative to the baseline models without preprocessing or tuned parameters. The models were tested using the methods and metrics presented in the previous chapter.

Models' performance for the target variables Score A, B, C, D, and E after therapy:

	Score A		Score B		Score C		Score D		Score E	
Model	**RMSE train**	**RMSE test**	**RMSE train**	**RMSE test**	**RMSE train**	**RMSE test**	**RMSE train**	**RMSE test**	**RMSE train**	**RMSE test**
Lasso	4.42	3.47	4.43	3.98	4.92	5.05	6.68	6.53	3.50	3.13
Ridge	0.60	5.17	0.73	7.00	0.69	7.07	1.62	12.95	0.52	4.73
Elastic Net	3.94	3.54	4.14	4.00	4.42	5.01	6.30	6.28	3.17	3.07
K-Nearest Neighbor	5.37	4.95	5.13	4.70	5.63	5.26	8.93	7.63	7.01	5.52
Support Vector Machines	15.00	8.94	15.80	10.59	26.14	18.52	21.23	17.26	14.95	12.32
Decision Tree	0.00	3.69	0.00	5.08	0.00	6.67	0.00	9.21	0.00	5.18
Random Forest	2.51	4.03	2.45	4.11	2.78	4.96	3.97	6.93	2.39	4.05
Gradient Boosting	0.03	3.89	0.04	4.29	0.04	5.76	0.10	7.43	0.04	4.46
X Gradient Boosting	0.00	3.59	0.00	5.65	0.00	5.71	0.02	6.76	0.00	5.13
Bagging Elastic Net	4.07	3.58	4.31	4.11	4.88	4.94	6.85	6.52	3.21	3.12
AdaBoost	3.18	4.65	1.58	4.94	2.04	5.45	2.81	7.24	1.64	3.88
Stacking	4.55	3.85	4.03	3.96	3.62	4.96	8.93	7.09	3.62	3.29
Multi-Layer Perceptron	61.50	59.9	15.36	12.44	15.74	13.49	15.60	13.14	12.61	11.15
Transformed Target XGB	0.02	3.48	0.03	5.17	0.01	6.17	0.01	6.76	0.01	5.92

Table 3: Models Comparison

In the next table, it is possible to observe the results of the most important target variable, the global Score, as it represents the general condition of the patients after the therapy:

	Global Score	
Model	**RMSE train**	**RMSE test**
Lasso	3.96	3.42
Ridge	0.75	6.41
Elastic Net	3.75	3.67
K-Nearest Neighbor	4.75	4.26
Support Vector Machines	15.89	11.50
Decision Tree	0.00	4.96
Random Forest	2.08	3.86
Gradient Boosting	0.05	3.88
X Gradient Boosting	0.00	5.05
Bagging Elastic Net	3.91	3.64
AdaBoost	1.75	4.36
Stacking	3.80	3.94
Multi-Layer Perceptron	12.87	11.20
Transformed Target XGB	0.01	5.78

Table 4: Global Score Models Comparison

5.1.1. Discussion

In both tables Table 3 and Table 4, it was possible to observe that Regularization models and Ensembles were the ones that provided the most powerful results. In part, these results were expected as Ensembles were the preferred models in many use cases using medical datasets, particularly rare disease datasets, and consequently with few data available [5]. Regularization models were also unsurprisingly effective given the dataset's nature, which presents linear relationships, as observed in the Data Understanding and Exploration section. Besides that, one should notice that regularization models such as Ridge and LASSO do not present overfitting or even present better scores for test data than for the training data. In contrast, other models are better at predicting the training data than the test. On the other hand, most Tree-based models, even in ensembles, were prone to overfitting, such as Decision Tree and Gradient Boosting, where train errors are significantly lower than test errors.

Surprisingly, Artificial Neural Networks (Multi-Layer Perceptron) and Support Vector Machines carry the worst results. In previous studies using Machine Learning in rare diseases, it was concluded that these models were on top of the models used [5]. Specially Support Vector Machines, given its effectiveness when dealing with high dimensional data and when the number of variables is greater than the number of observations. Artificial Neural Networks are only capable of providing satisfactory

results when having a large quantity of data, in this way, there is a cause for the results being significantly poor [32]. On the other hand, in [20] it was shown that both models tend to have a poor generalization capability when dealing with imbalanced datasets. For these reasons, the mentioned models were excluded from the current study. Besides being black box models and requiring a lot of computational power and time to train, they did not even provide close to good results.

In all scenarios, the results were worsened by the Transformed Target, which reshaped the target's distribution for it to look more Gaussian-like. This could be due to the fact that when the function to reshape the distribution is applied, the relationships between the variables are not preserved. In this way, the linear relationship between the independent and target variables is lost.

Chained models did not improve the results as well. After predicting a score variable and feeding it to the next model, this variable ended up not being chosen in the Feature Selection process, not adding any useful information to the model. Besides that, the best-performing model is not the same for every variable, which complicates the learning schema.

5.2. Final Models

It becomes extremely difficult to enhance the results obtained by the baseline models when having small datasets. Nonetheless, in search of better scores, the procedure was to exploit the weaknesses of each model. For example, to enhance the Decision Trees scores, the best approach was pruning and applying ensembles. In this way, it was possible to prevent them from memorizing the training data, avoiding overfitting. Moreover, the best scores for the Regularization models were accomplished by increasing the λ value, again preventing overfitting.

Therefore, the best models were chosen after tuning the preprocessing techniques and the models' parameters. The final selection was made considering Occam's razor principle, which states that when facing two models with equivalent performance, one should prefer the simpler model with fewer assumptions, capable of generalizing better.

5.2.1. Final Results

In this section, the final results for each target variable are presented individually. The results include each model's best techniques and parameters and their respective scores. Besides that, it is also possible to visualize the most significant variables for each model according to the average of 41 repeated Leave-One-Out Cross-Validation. The obtained scores of the Genetic Programming research [8] are also provided. It was possible to directly compare the results against the Genetic Programming since the predictive power of both researches was computed using the same evaluation method and metrics.

Score A

Technique	Value	
Binning:	K-Means	
Transformations:	Log, Power, Quantile	
Scaler:	Robust	
Feature Selection:	Rank Selector	
Data Augmentation:	SMOGN	
Model:	Ridge	
Parameters:	λ	2.2
	Solver	Svd

Table 5: Final Model and Techniques for Score A

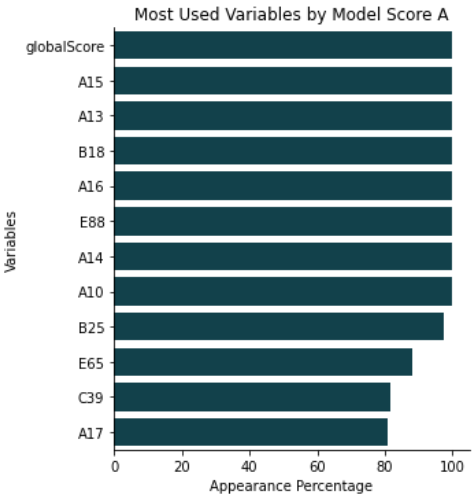

Figure 10: Feature Importance for Score A

Metric	Performance
$\overline{\text{RMSE}}$ train	3.64 ± 0.45
$\overline{\text{RMSE}}$ test	3.18 ± 4.28
$\overline{\text{MAPE}}$ train	$2.89\% \pm 0.39$
$\overline{\text{MAPE}}$ test	$4.03\% \pm 5.78$
GP $\overline{\text{RMSE}}$ test	3.83 ± 4.36

Table 6: Final Results variable Score A

Figure 11: Final Results variable Score A

For the target variable *Score A* the final leading model was Ridge Regression. Since the variables from the scores A had outliers (as observed in Data Understanding and Exploration) and Ridge Regression is sensitive to them, Robust scaler was applied. This method is well known for its capability of reducing the impact of extreme values. Besides that, Ridge Regression is not able to perform feature selection as efficiently as LASSO, which throws away non-important variables. In this way, the performance of this model was improved by applying the Rank Selector method. This is the most powerful feature selection method since it aggregates information from several techniques and compensates the Ridge disadvantage of not being able to discard irrelevant features. Moreover, increasing the λ value also made it possible for the model to better generalize to the test data.

Score B

Technique	Value	
Binning:	K-Means	
Transformations:	Log, Power, Quantile	
Scaler:	-	
Feature Selection:	F-statistic	
Data Augmentation:	SMOGN	
Model:	LASSO	
Parameters:	λ	1.0

Table 7: Final Model and Techniques for Score B

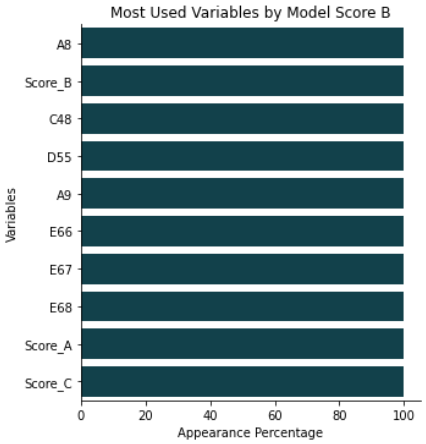

Figure 12: Feature Importance for Score B

Metric	Performance
$\overline{\text{RMSE}}$ train	4.73 ± 0.16
$\overline{\text{RMSE}}$ test	3.89 ± 4.29
$\overline{\text{MAPE}}$ train	$3.86\% \pm 0.19$
$\overline{\text{MAPE}}$ test	$5.11\% \pm 5.76$
GP $\overline{\text{RMSE}}$ test	4.54 ± 4.89

Table 8: Final Results variable Score B

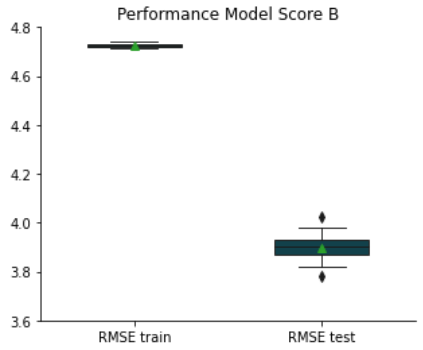

Figure 13: Final Results variable Score B

Until now, Ridge and LASSO were considered the best models for the first target variables. This can be explained by the linear relationship already observed in the Data Understanding and Exploration chapter. Besides that, F-statistic was used to boost the scores of LASSO. F-statistic quantifies the linear relationship between the variables, providing to LASSO the strictly necessary information to get the best results. Although not visible in the Figure 12 that represents the top 10 of variables used by the model, the binning technique was also applied to enhance the scores of this model. Variable binning, also known as variable discretization, often improved the models' results, since this method provides an easy way of treating possible outliers and simplifies the models learning.

Score C

Technique	Value	
Binning:	-	
Transformations:	Log, Power, Quantile	
Scaler:	Robust	
Feature Selection:	LASSO	
Data Augmentation:	-	
Model:	Stacking	
Parameters:	estimators	Ridge, Elastic Net, Random Forest
	final_estimator	Linear Regression

Table 9: Final Model and Techniques for Score C

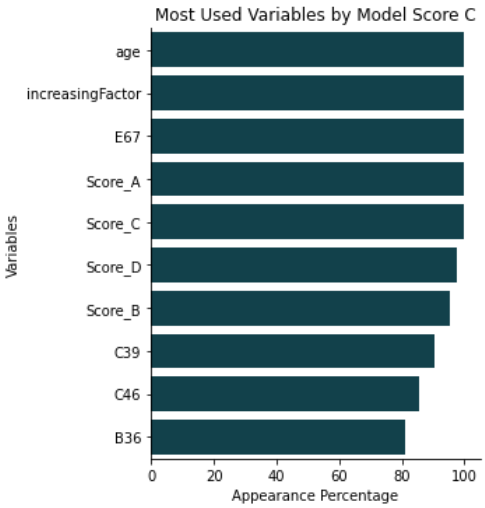

Figure 14: Feature Importance for Score C

Metric	Performance
$\overline{\text{RMSE}}$ train	3.06 ± 0.18
$\overline{\text{RMSE}}$ test	4.43 ± 4.94
$\overline{\text{MAPE}}$ train	$3.87\% \pm 0.40$
$\overline{\text{MAPE}}$ test	$7.99\% \pm 11.47$
GP $\overline{\text{RMSE}}$ test	4.71 ± 4.94

Table 10: Final Results variable Score C

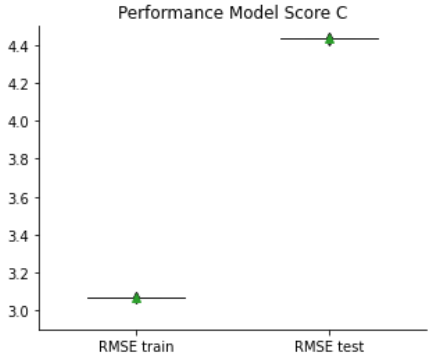

Figure 15: Final Results variable Score C

A widely used strategy when individual models are not achieving the desired results, is to put them together in a "stack" and then use another final model to improve their individual predictions. In this way, stacking was the winning model for the variable *Score C*. The stacking consists of an ensemble of 3 models referred above and uses a Linear Regression as the final model, considering that in the end it is recommended to use the simplest model to reduce not only the time and computational power required, but also the non-disruption of the predictions of each individual model, in order to not reduce their strength. The remaining details about the models' parameters can be found in the previous chapter.

Score D

Technique	Value	
Outliers:	Clipping	
Binning:	-	
Transformations:	Log, Power, Quantile	
Scaler:	-	
Feature Selection:	F-statistic	
Data Augmentation:	SMOGN	
Model:	LASSO	
Parameters:	λ	1.5
	Selection	Random
	Tol	0.001

Table 11: Final Model and Techniques for Score D

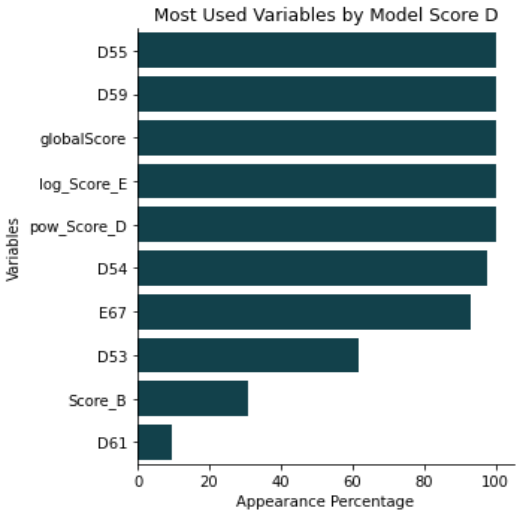

Figure 16: Feature Importance for Score D

Metric	Performance
$\overline{\text{RMSE}}$ train	7.93 ± 0.22
$\overline{\text{RMSE}}$ test	5.84 ± 7.61
$\overline{\text{MAPE}}$ train	$9.53\% \pm 0.45$
$\overline{\text{MAPE}}$ test	$11.14\% \pm 13.63$
GP $\overline{\text{RMSE}}$ test	5.93 ± 6.23

Table 12: Final Results variable Score D

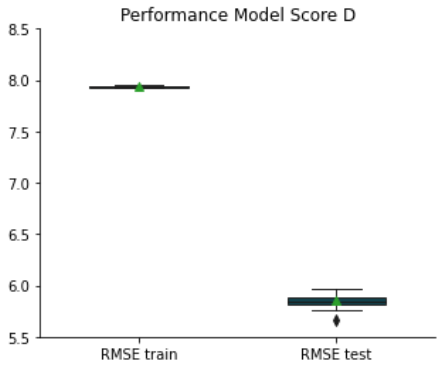

Figure 17: Final Results variable Score D

As already mentioned, linear models are sensitive to outliers. Clipping was the best solution found for dealing with the outliers and improve the scores of the winning model, LASSO, along with the transformations applied to the variables to treat skewness, such as the logarithm and power functions (the functions can be seen in the feature importance's chart). Besides that, it was used F-statistic for feature selection, that as mentioned before is the most appropriated feature selection for linear models. Regarding the model's final performance, when comparing the results in the variable Score A and B with the ones obtained in the *Score D*, the former were better. This could be easily inferred as the former are the simplest exercises to complete (lying and rolling) comparing with D, that represents standing exercises, like raising a foot or squatting.

Score E

Technique	Value	
Binning:	-	
Transformations:	Log, Power, Quantile	
Scaler:	-	
Feature Selection:	LASSO	
Data Augmentation:	-	
Model:	Bagging of Elastic Net	
Parameters:	n_estimators	20
	λ	2.0
	selection	random

Table 13: Final Model and Techniques for Score E

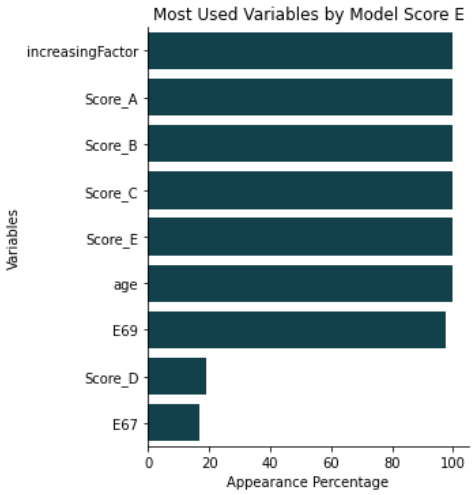

Figure 18: Feature Importance for Score E

Metric	Performance
$\overline{\text{RMSE}}$ train	3.43 ± 0.13
$\overline{\text{RMSE}}$ test	3.05 ± 2.78
$\overline{\text{MAPE}}$ train	$5.96\% \pm 0.23$
$\overline{\text{MAPE}}$ test	$7.22\% \pm 6.22$
GP $\overline{\text{RMSE}}$ test	2.95 ± 2.84

Table 14: Final Results variable Score E

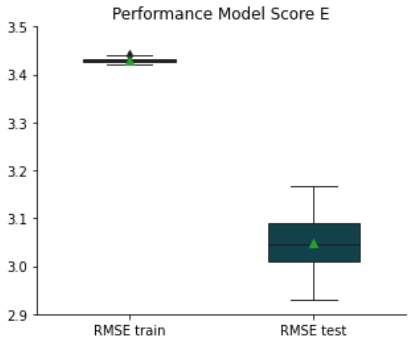

Figure 19: Final Results variable Score E

By looking at the graphs of *Scores A, D,* and *E*, it is clear that, when using regularization methods, the train errors are greater than the test errors. Given the nature of the dataset, which presents some outliers, and given the capability of these models to introduce less variance in exchange of more bias, this situation ends up being regular. Regarding the feature selection technique, as Elastic Net is a wrapper method, it has already implemented a system of automatic feature selection, so it did not need further variable selection techniques. When applying Elastic Net within the Bagging technique, "Wisdom of the Crowd" theory has been proven to increase the results of an individual model. On the other hand, when attempting to apply Data Augmentation methods to increase the dataset size, the results only got worse. This could be explained by the fact that not all the models are enhanced when applying data augmentation techniques. This happens mainly when dealing with small datasets, as it was shown in [26] that used clinical datasets to study the impact of data augmentation, concluding that these techniques on small datasets are less useful than for larger ones.

Global Score

Technique	Value	
Binning:	-	
Transformations:	Log, Power, Quantile	
Scaler:	Robust	
Feature Selection:	Mutual Information	
Data Augmentation:	SMOGN	
Model:	Random Forest	
Parameters:	split_criterion	mse
	max_depth	8
	min_samples_split	2

Table 15: Final Model and Techniques for Global Score

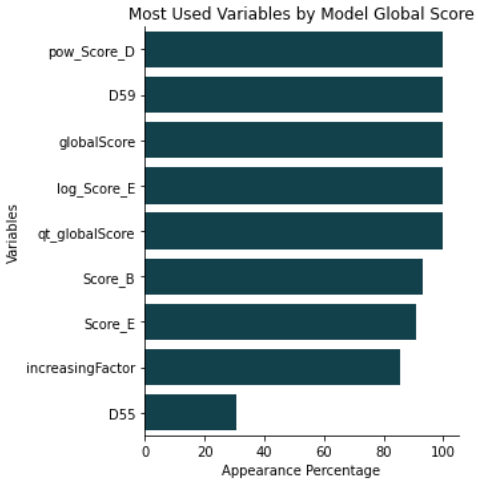

Figure 20: Feature Importance for Global Score

Metric	Performance
$\overline{\text{RMSE}}$ train	2.64 ± 1.16
$\overline{\text{RMSE}}$ test	2.87 ± 3.35
$\overline{\text{MAPE}}$ train	$2.85\% \pm 1.43$
$\overline{\text{MAPE}}$ test	$4.37\% \pm 5.14$
GP $\overline{\text{RMSE}}$ test	3.78 ± 3.91

Table 16: Final Results variable Global Score

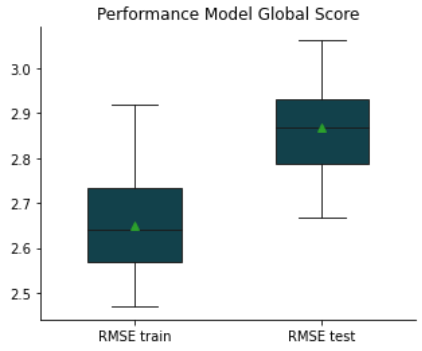

Figure 21: Final Results variable Global Score

Not surprisingly, Random Forest presented the best scores to predict the final variable Global Score, after the therapy. As suggested in several studies in [2], Random Forests tend to have a great performance when dealing with rare disease datasets since ensembles can balance individual over-learning. Although outliers do not usually have an exceptional influence on Random Forests, the Robust scaler increased its results, along with MSE for the split criterion that squares the errors, ending up giving more importance to greater errors. To control the overfitting, the maximum depth allowed in the Decision Trees was diminished. To also prevent overfitting, the best feature selection technique was chosen, which was mutual information, which reduced the needed information for the model to only 15 variables. In this way, for future predictions, one could only provide to the model that information. It is also important to note that almost all the models, including the present one, enriched their results after applying SMOGN for data augmentation.

Regarding the final results, the standard deviation presented is significantly high. Given the dataset's nature, which registers skewed variables and some outliers, the variance of the results ends up being high. However, the MAPE score was around 4%, meaning that, on average, the distance from the actual value is only 4%, which reveals that the model was surprisingly effective in predicting the test data. The RMSE is approximately 3, which can be considered acceptable on a scale from 0 to 100.

5.2.2. Discussion

The scores obtained by the presented models were similar to the results obtained by the Genetic Programming presented in [8]. Nevertheless, several models have outperformed the results of the Genetic Programming, even without any preprocessing and model tuning, as it is the case of the Global Score variable, which is the most important target variable. Besides that, Genetic Programming has the main disadvantage of not being totally interpretable if presenting a complex final model, and the models developed for the current research are in fact, easy to understand, which is an important aspect when facing medical problems. Moreover, Occam's razor principle states that one should prefer simpler models over complex ones [17], and the presented models are mainly known for their simplicity.

5.2.3. Models Interpretability

As mentioned previously, besides considering the models' predictive performance, it is also critical to consider their interpretability. This aspect is of extreme importance, particularly when dealing with high-risk decisions, where it is crucial to comprehend the model's structure and its dependencies [22]. Although the categories of the models were already chosen considering this, it is also possible to apply techniques such as the Ceteris-Paribus to introduce interpretability to the models. With Ceteris-Paribus, it is possible to quantify how much each variable influences the model's final prediction, allowing for its further understanding. The next figures represent the Ceteris-Paribus values for two variables belonging to the final global model. It was necessary to choose a patient to apply the Ceteris-Paribus principle randomly, "other things held constant", which means while keeping all the other variables values constant, the variable in analysis is modified to evaluate its impact on the target variable.

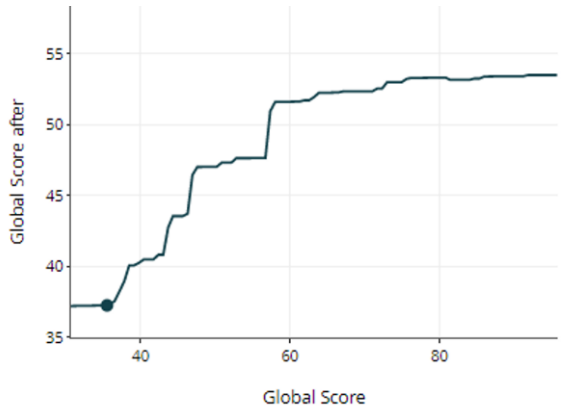

Figure 22: Ceteris-Paribus for variable Global Score

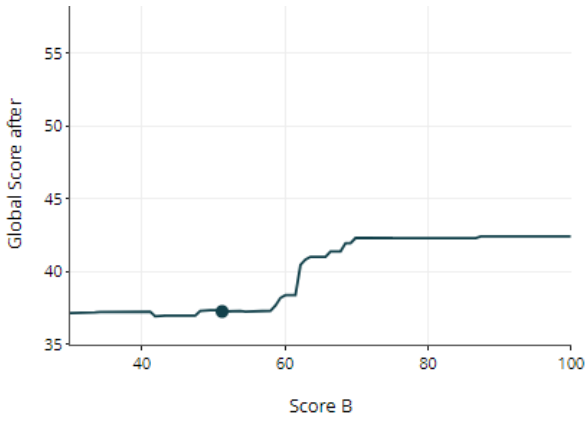

Figure 23: Ceteris-Paribus for variable Score B

The ball marker represents the actual value of a random patient chosen for the variable in the analysis, and the line represents the calculated Ceteris-Paribus values. With these visualizations, it is possible to consider which exercises could be applied to increase the mobility score of the patient since it is possible to have a personalized overview of how each variable impacts the final overall score of the patient.

5.2.4. Answer to Research Questions

Having the best models designed and constructed, it is finally possible to answer the research questions defined in the beginning of the project. In this way, the following research questions were raised with the goal of better understanding the overall therapy results, broadly speaking:

A. Is the therapy being effective for the patients?

As observed in the Data Exploration chapter, on average, the patients' mobility scores increase after the therapy, verifying that in a general way the therapy is being efficient. By all the means, the final results depend on each patient individually and their correspondent condition, but it is possible to conclude that the therapy on average has a positive impact on the patients' motor functions.

B. Which are the exercises that have the highest impact on the therapy result?

By observing the feature importance's of the last model, one can conclude that the most important exercises that have the highest influence on the therapy result are the exercises from the section D.

After broadly considering the effects of the therapy, questions focused on a specific patient may also be considered in order to better understand the outcomes of an individual patients' treatment:

C. Is the patient going to respond to the treatment?
D. How much mobility is the patient going to recover?
E. Which functional motions will the patient improve?
F. Is it worth to allocate efforts to the patient at the moment?

To answer all the previous questions, one may use the constructed models and analyze their predictions. To accomplish this, it is necessary to present to the model the historical data of the patient in analysis. An example can be seen in the following table, which presents data about a fictional patient:

Age	Diagnosis	# Protocols		Score A	Score B	Score C	Score D	Score E	Global Score
8	Pallister Killian Syndrome	1	…	52.94	46.25	25.0	34.62	30.21	46.73

Then, one can apply the developed models or can choose just one model depending on the area of interest (whether A, B, C, D, E or global). After all, if only an overall perspective on how the patient is going to react to the treatment is desired, one should use the model that predicts the Global Score. The patient data is then feed to the final model, which in turn predicts that the patient mobility scores. In the next table, it is possible to observe the predictions returned by the best model of each Score variable, for the patient in analysis:

Score A after	Score B after	Score C after	Score D after	Score E after	Global Score after
63.06	58.90	59.54	59.12	59.40	57.07

By comparing the scores prior to the therapy with the ones predicted after the therapy, it is possible to conclude that the movements from area C were the motor functions that improved the most, and the least from area A. It is also possible to observe that the model predicted an increase on the overall patient mobility from 46.73 to 57.07. Based on the model's prediction, this patient would respond positively to the therapy applied, more precisely the therapy would be highly effective and worthwhile, resulting in an improvement of 22% in the patient's motor skills.

Last but not the least:

G. Are at least some Machine Learning methods able to learn reliable and robust models even in presence of such small data and different conditions?

Many techniques had to be explored and adapted to face the major challenge of rare diseases: lack of data. Nonetheless, as it was possible to observe in the previous chapters, most of the models had surprisingly good results, being the best models capable of making predictions with an error rate of only 4%, providing a feasible and credible way for medical specialist to know beforehand how the patient would react to the therapy, giving support in future medical decisions.

6. CONCLUSIONS

In this last chapter, it is possible to find a brief summary of the work developed during this research and its respective conclusions. The limitations encountered during the research are also present, as well as some recommendations for possible future work.

6.1. SUMMARY

Rare diseases are considered a public health issue, given that approximately 7% of all hospitalizations are linked to a rare disease. Despite being rare, they directly or indirectly affect a significant part of the population. However, one of the main disadvantages when investigating rare diseases is the lack of information and investment around this topic, considering that each individual disease is rarely found. Besides that, the astounding difference between the diseases and the patients leads to difficulties in their diagnosis and treatment, demanding highly personalized medical care.

These types of situations require computerized support systems. In this way, Artificial Intelligence and Machine Learning models can be the solution to simplify the work of clinicians and medical experts, which is becoming burdensome in this environment. Based on a literature review, most existing models around this topic suggest satisfying preliminary results. Although there has indeed been progress concerning this issue, there is still much to do.

On these grounds, to take advantage of Machine Learning characteristics in rare diseases, this research aimed to identify a model capable of predicting the improvement of patients' motor functions after attending physiotherapy sessions. This model was built with the principal goal of predicting if a rare disease patient would respond to the treatment or not, providing a basis for prognostic guidance for planning clinical treatment. Several Machine Learning models were explored and tested in an attempt to accomplish the final goal, resorting to clinical data of patients provided by the Raríssimas organization. The models and techniques were chosen taking into consideration this particular use case, which demanded human-readable to provide a smoother connection between the clinicians and Machine Learning. The models had also to adapt to completely new data, given the difference between patients, which required models capable of generalizing. They also had to be robust and capable of mitigating the risks of having small datasets. The final models that presented the most promising results and covered all the requirements were regularization models and ensembles. Additionally, the models resorted to techniques for handling data quality issues without deleting any data. Moreover, data augmentation techniques were also part of the final solution to surpass the main disadvantage of rare diseases: the lack of data.

By predicting the improvement of patients with complex neurological disorders, it is possible to know beforehand whether the patient will respond or not to the treatment. Identifying and distinguishing between a respondent and a non-respondent patient is crucial nowadays, in order to save effort and costs. Besides that, they can support better medical decisions, simplifying and accelerating the involved processes currently done manually.

6.2. LIMITATIONS AND FUTURE RESEARCH

While developing the final solution, several problems were identified. However, the major limitation of this study was the lack of data. Future work could be focused on this challenge and merge different data sources to construct a more effective model. For this research, the chosen approach to solve this issue was data augmentation, which may not be the most appropriate solution since it only generates artificial data. Further documented historical data in Raríssimas could also leverage the results.

If in future research, the scores obtained by the presented models are surpassed, then it is possible to increase them further by resorting to chained models. These models were already attempted, although as the predictions were not highly accurate, this type of model was discarded. If better predictions are obtained, chained models may provide better scores since there is a relationship between the target variables.

Besides that, different evaluation metrics for the models should be considered since RMSE does not consider imbalanced datasets. Although metrics like RMSE and MAPE are highly popular and are mostly used for comparison purposes, this can be misleading for studies involving small and imbalanced datasets. Furthermore, evaluation metrics are not enough to thoroughly test the models. The final models should be tested against medical experts to prove their value and accuracy. For these reasons, this topic deserves further investigation.

7. BIBLIOGRAPHY

[1] European Comission, "European Comission - EU research on rare diseases," 10 1 2022. [Online]. Available: https://research-and-innovation.ec.europa.eu/research-area/health/rare-diseases_en.

[2] S. Brasil, C. Pascoal, R. Francisco, V. D. R. Ferreira, P. A. Videira and G. Valadão, "Artificial Intelligence (AI) in Rare Diseases: Is the future brighter?," *Genes,* vol. 10, no. 12, p. 24, 2019.

[3] Raríssimas – Associação Nacional de Deficiências Mentais e Raras, "Raríssimas," [Online]. Available: https://rarissimas.pt/adoencarara/. [Accessed 18 07 2022].

[4] SIOP Europe the European Society for Paediatric Oncology, "Rare Diseases," [Online]. Available: https://siope.eu/activities/european-advocacy/rare-diseases/. [Accessed 10 January 2022].

[5] J. Schaefer, M. Lehne, J. Schepers, F. Prasser and S. Thun, "The use of machine learning in rare diseases: a scoping review," *Orphanet Journal of Rare Diseases,* p. 10, 2020.

[6] A. Rajkomar, J. Dean and I. Kohane, "Machine Learning in Medicine," *The New England Journal of Medicine,* p. 12, 2019.

[7] S. Ronicke, M. C. Hirsch, E. Türk, K. Larionov, D. Tientcheu and A. D. Wagner, "Can a decision support system accelerate rare disease diagnosis? Evaluating the potential impact of Ada DX in a retrospective study," *Orphanet Journal of Rare Diseases,* p. 12, 2019.

[8] I. Bakurov, M. Castelli, L. Vanneschi and M. J. Freitas, "Supporting Medical Decisions for Treating Rare Diseases through Genetic Programming," *Springer Verlag,* p. 17, 2019.

[9] E. Scheeren, L. P. G. Mascarenhas, C. Chiarello and A. C. M. S. Costin, "Description of the Pediasuit ProtocolTM," *Fisioterapia em Movimento,* p. 9, 2012.

[10] McMaster University, "CanChild Resources - Gross Motor Function Measure (GMFM)," [Online]. Available: https://canchild.ca/en/resources/44-gross-motor-function-measure-gmfm. [Accessed 31 07 2022].

[11] D. T. Larose and C. D. Larose, Data Mining and Predictive Analytics – Wiley Series on Methods and Applications in Data Mining, 2015.

[12] L. Torgo, R. P. Ribeiro, B. Pfahringer and P. Branco, "SMOTE for Regression," in *Progress in Artificial Intelligence*, 2013.

[13] M. S. Santos, J. P. Soares, P. H. Abreu, H. Araujo and J. Santos, "Cross-Validation for Imbalanced Datasets: Avoiding Overoptimistic and Overfitting Approaches," *IEEE Computational intelligence magazine,* p. 18, 2018.

[14] S. S. Lee, "Regularization in skewed binary classification," *Computational Statistics,* p. 16, 1999.

[15] P. Branco, R. P. Ribeiro and L. Torgo, "UBL: an R Package for Utility-Based Learning," p. 88, 2016.

[16] P. Branco, L. Torgo and R. P. Ribeiro, "SMOGN: a Pre-processing Approach for Imbalanced Regression," in *1st International Workshop on Learning with Imbalanced Domains - Theory and Applications*, 2017.

[17] M. Kuhn and K. Johnson, Applied Predictive Modelling, Springer, 2013.

[18] R. Tibshirani, T. Hastie and J. H. Friedman, "Regularized Paths for Generalized Linear Models Via Coordinate Descent," *Journal of Statistical Software,* vol. 33, no. 1, p. 22, 2010.

[19] J. Cai, J. Luo, S. Wang and S. Yang, "Feature selection in machine learning: A new perspective," *Neurocomputing,* vol. 300, p. 10, 2018.

[20] M. Schubach, M. Re, P. N. Robinson and G. Valentini, "Imbalance-Aware Machine Learning for Predicting Rare and Common Disease-Associated Non-Coding Variants," p. 15, 2017.

[21] L. Torgo, B. Krawczyk, P. Branco and N. Moniz, "Evaluation of Ensemble Methods in Imbalanced Regression Tasks," p. 12, 2017.

[22] A. Lemanska-Perek, K. Kobylinska, P. Biecek, T. Skalec, M. Tyszko, W. Gozdzik and B. Adamik, "Explainable Artificial Intelligence Helps in Understanding the Effect of Fibronectin on Survival of Sepsis," *Vells,* p. 13, 2022.

[23] M. Tschuggnall, V. Grote, M. Pirchl, B. Holzner, G. Rumpold and M. J. Fischer, "Machine learning approaches to predict rehabilitation success based on clinical and patient-reported outcome measures," *Informatics in Medicine Unlocked,* p. 10, 2021.

[24] I. Duran, C. Stark, A. Saglam, A. Semmelweis, H. L. Wunram, K. Spiess and E. Schoenau, "Artificial intelligence to improve efficiency of administration of gross motor function assessment in children with cerebral palsy," *Developmental Medicine & Child Neurology,* p. 7, 2021.

[25] S. Decherchi, E. Pedrini, M. Mordenti, A. Cavalli and L. Sangiorgi, "Opportunities and Challenges for Machine Learning in Rare Diseases," 2021.

[26] J. Beinecke and D. Heider, "Gaussian noise up-sampling is better suited than SMOTE and ADASYN for clinical decision making," *BioData Mining,* p. 11, 2021.

[27] M. S. Santos, P. H. Abreu, H. Araújo and J. Santos, "Cross-Validation for Imbalanced Datasets: Avoiding Overoptimistic and Overfitting Approaches," *IEEE Computational intelligence magazine,* p. 18, 2018.

[28] C. Faviez, X. Chen, N. Garcelon, A. Neuraz, B. Knebelmann, R. Salomon, S. Lyonnet, S. Saunier and A. Burgun, "Diagnosis support systems for rare diseases: a scoping review," *Orphanet Journal of Rare Diseases,* p. 16, 2020.

[29] H. Liu, H. Jiang, X. Wang, J. Zheng, H. Zhao, Y. Cheng, X. Tao, M. Wang, C. Liu, T. Huang, C. Jin, X. Li, H. Wang and J. Yang, "Treatment response prediction of rehabilitation program in children with cerebral palsy using radiomics strategy: protocol for a multicenter prospective cohort study in west China," *Quantitative Imaging in Medicine and Surgery,* p. 11, 2019.

[30] L. Vanneschi, I. Bakurov and M. Castelli, "An Initialization Technique for Geometric Semantic Genetic Programming based on Demes Evolution and Despeciation," in *IEEE Congress on Evolutionary Computation*, 2017.

[31] S. Brasil, C. J. Neves, T. Rijoff, M. Falcão, G. Valadão, P. A. Videira and V. d. R. Ferreira, "Artificial Intelligence in Epigenetic Studies: Shedding Light on Rare Diseases," *Frontiers in Molecular Biosciences,* p. 14, 2021.

[32] R. Andonie, "Extreme Data Mining: Inference from Small Datasets," *International Journal of Computers, Communications & Control,* 2010.

[33] M. H. Schwartza and M. E. Munger, "Using machine learning to overcome challenges in GMFCS level assignment," p. 6, 2018.

[34] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher and M. Perrot, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research,* p. 6, 2011.

[35] D. H. Jacqueline Beinecke, "Gaussian noise up-sampling is better suited than SMOTE and ADASYN for clinical decision making," *BioData Mining,* p. 11, 2021.