Exploiting machine learning techniques to predict Alzheimer's Disease Progression

João Pedrosa
joao.paulo.pedrosa@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

January 2021

Abstract

Affecting 30% of the population above 85 years of age, Alzheimer's Disease (AD) is a chronic neurodegenerative disease responsible for about two-thirds of dementia cases worldwide. With the ageing in the global population, the number of AD patients is expected to rise significantly in the coming years. As most of the tests that detect AD are either too expensive or invasive, we turn our heads to neuropsychological tests and machine learning to help solve this issue. These tests assess the cognitive abilities of patients and can be taken in less than one hour with little expense. In search for the optimal solution, we look into state-of-the-art technologies for classification, missing value imputation (MVI) and other steps in the data mining process. From this, we manage to build a working classification pipeline capable of analyzing the test data and predicting a patient's future conversion to AD as well as its time frame. This prediction is performed for a certain time window, and with a certain degree of confidence. Our solution to improve upon this work is to implement state-of-the-art algorithms and test different configurations until an ideal setup is determined.

Keywords: Alzheimer's Disease; Mild Cognitive Impairment; Machine Learning; Deep Learning; Classification; Missing Value Imputation; Class Imbalance; Feature Selection.

1. Introduction

Alzheimer's Disease (AD) is a chronic neurodegenerative disease responsible for about two thirds of dementia cases worldwide. Its causes are poorly understood and it remains untreatable. As of 2019 it is estimated that AD affects 3% of people aged 65 or younger, although the number rises to above 30% for people aged 85 and older [1].

At the earliest stages of the disease a full manifestation of the symptoms does not occur. Usually, a patient is first diagnosed with Mild Cognitive Impairment (MCI) when some cognitive declines begin to appear (such as memory lapses and motor difficulties). However, diagnosing early stage AD is not a straightforward process, as it is necessary to understand the differences between MCI caused declines and declines generally caused by ageing. Furthermore, neurodegenerative diseases can take years to manifest, all the while draining at MCI patients' cognitive abilities. By the time a patient is diagnosed with dementia, their brain would have already suffered irreparable damage, severely impacting autonomy and cognition. The World Alzheimer Report 2015 [12] estimated that the number of people living with dementia will nearly triple to 131 million by 2050.

Being able to predict the progress from patients with MCI to AD is of great importance. It can help with the appropriate selection of therapeutic interventions for each unique patient, as well as improving their quality of life for the following years by slowing the decline in cognitive skills. It can also have a great social-economic impact, as it would reduce unnecessary tests and procedures on millions of patients worldwide. Lastly it could have a large impact on the families and patients themselves, by providing some predictability to a disease ridden with uncertainty.

This work addresses the problem of predicting the evolution from MCI to AD in any given patient, as well as predicting the time of such evolution. To achieve this, several datasets with neuropsychological data will be used. The datasets contain the scores of multiple neuropsychological tests performed by each patient to evaluate their present cognitive capabilities. The different sets are organized into yearly intervals, so as to help determine how long the disease takes to evolve in each patient.

We will use state of the art machine learning techniques to help determine the time until conversion to AD, with a particular focus on the field of missing value imputation. Our aim is to innovate and explore new and unique approaches to solve the problems at hand.

2. Background

The CCC, a study conducted by researchers from the *Universidade de Medicina de Lisboa* (or Faculty of Medicine of the University of Lisbon) (FMUL), was launched to evaluate the progression of MCI and AD on Portuguese patients [14]. The study admitted patients initially diagnosed with MCI and had them take BLAD tests. These were performed and evaluated with the help of several institutions (Laboratory of Language Studies at Santa Maria Hospital and a Memory Clinic, both in Lisbon, and the Neurology Department at University Hospital in Coimbra). After their initial tests, the patients had follow-up appointments where they were tested for dementia.

The initial dataset (CCC-1) with the patients' test scores and diagnosis was released on April 2017. Since this release, and as of present day, the dataset has been updated twice, with CCC-2 being released in October2017 and CCC-3 being released in October 2018. The dataset is divided into two subsets: one for patients recruited in Lisbon and another for patients in Coimbra.

The original datasets were organized according to a First-Last approach, where one single dataset would hold the data for all patients and each data instance had information regarding their initial and last evaluation. In [11], T. Pereira proposed reorganizing the dataset into a Time Window approach. This approach groups data according to a specific temporal frame. For example, the dataset referring to the five year window would have the initial tests performed by each patient and a class indicating whether they converted to dementia after that five year interval.

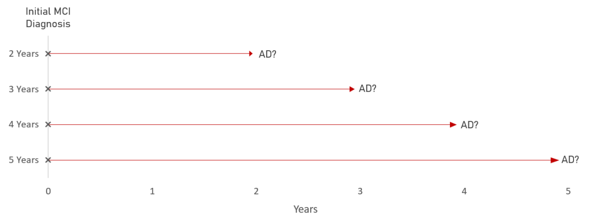

Figure 1: Representation of the four datasets

For this work we utilized CCC-3 following the Time Windows approach, as proposed by T. Pereira. The data is organized into four different sets with each referring to a different time window (2, 3, 4 and 5 years). The characterization of these datasets is presented in tab:Datasets.

The datasets contains a heterogeneous set of data. This includes both categorical and numerical features, as well as features with different proportions of missing data. Each feature has a different or unknown correlation to MCI and AD. In order to solve problems like these with such datasets there is the need to set up a classification pipeline. Ideally, any step in this pipeline should be adjustable without requiring any changes to the remaining steps. This structure allows us to experiment with different methods and different configurations in order to achieve the best possible results. In problems like these, the pipeline is generally composed of the following steps:

Table 1: Characterization of the datasets utilized: Number of stable MCI cases (sMCI), number of converter MCI cases (cMCI), total number of samples and percentage of missing data.

Dataset		sMCI	cMCI	Total	Missing %
Lisbon	2Y	394	107	501	28.38%
	3Y	305	163	468	28.27%
	4Y	227	204	431	28.34%
	5Y	175	235	410	28.96%
Coimbra	2Y	64	10	74	32.72%
	3Y	50	17	67	32.46%
	4Y	40	21	61	32.64%
	5Y	30	23	53	33.89%

2.1. Data Cleaning

This step consists in scanning the database to eliminate any errors or inconsistent data. Some examples can include the presence of decimal values in categorical features, or values that have no explanation other than human error. Features with an extremely high amount of missing data should also be eliminated. As for the datasets being used, this step has already been performed by reporting all errors to the doctors responsible for its maintenance.

2.2. Feature Selection

In the Feature Selection (FS) step, a subset of features are selected from the original dataset, based on how relevant or unique they are. This step provides significant benefits, such as reducing overfitting, decreasing training time and easing understanding. Several different methods can be used for this process, with each of them measuring and comparing different characteristics of the features. The considered FS methods for this paper are: Mutual Information Maximization (MIM) [16]; Chi-Squared (CS) [3]; And Recursive Feature Elimination with a Support Vector Machine (RFE SVM) [17].

2.3. Missing Value Imputation

In this step, algorithms are applied to the dataset in order to generate values to replace any missing data. The existence of missing data within a dataset can heavily affect the results of some classifiers, making this an important step in the classification process. There are several models to address MVI, which can be divided into two categories: Discriminative and Generative [10]. Discriminative Models merely model the decision boundaries between the different classes while Generative Models focus on learning the probability distribution of the individual classes. For this paper we selected the following Discriminative MVI methods: Overall Sample Mean (OSM); MissForest (MF) [15]; And Multiple Imputation by Chained Equations (MICE) [13]. We also selected the following Generative MVI methods: Denoising Autoencoder (DAE); Variational Autoencoder (VAE); And Generative Adversarial Network (GAN).

2.4. Handling Class Imbalance

A class imbalance exists when the final classifications for all samples in the dataset are not proportionally distributed. Most classifiers excel with a balanced dataset. To address this issue we can utilize class balancing methods, that can be classified as undersampling or oversampling methods. The process of undersampling can be simply described as the removal of instances belonging to the majority class of the dataset until class balance is reached, while oversampling consists of adding new (synthetic) data instances to the minority class. For this paper we considered the following undersamping methods: Random Undersampling (RU); Tomek Links [5]. As well as the following oversampling methods: Synthetic Minority Oversampling Technique (SMOTE) [4]; Adaptive Synthetic (ADASYN) [8]; MAHAKIL [2].

2.5. Classification

This is the process that takes our modified datasets and attempts to extract patterns from them. The process of assigning a class to an observation is described as classification, *i.e.*, through the application of algorithms that perform supervised learning on a set of data and then use the learned model to predict the classes of other data instances. The classifiers that are explored in this paper are the following: Naive Bayes (NB); Decision Trees (DT); Random Forests (RF); K-Nearest-Neighbour (KNN); Logistic Regression (LR); Support-Vector Machine (SVM); Neural Networks (NN).

2.6. Evaluation

In order to determine the best technologies to use and the overall best configuration, all results need to be compared and evaluated. Any model will have no value without a relevant accuracy assessment.

In order to compare the different MVI methods tested we selected the Root mean square error (RMSE) measurement, which quantifies the difference between expected and obtained values. And to compare the classification results we chose to use the following set of metrics: Accuracy; Sensitivity; Specificity; Area under ROC (ROC AUC).

The accuracy measure gives us the ratio of correct predictions to the size of the dataset, and is the easiest to understand. On the downside, accuracy returns less differentiated values than other metrics do. By grouping false positive and false negative values, it does not provide enough information on the classifier's behaviour. Sensitivity (or true positive rate) and specificity (or true negative rate) make up for this by providing more differentiated results and by presenting constant results regardless of the dataset being balanced or not. Finally the ROC AUC measure computes the sensitivity against the false positive rate $(1 - Specificity)$ so as to help compare the two measurements simultaneously.

3. Implementation

The experimental components of this paper are divided into two groups. The first deals with MVI methods and pits the selected methods against each other in order to determine the best ones. The second set deals with a complete data mining pipeline capable of classifying data instances from the CCC-3 dataset.

3.1. Missing Value Imputation

The MVI methods were implemented in python, with OSM, MICE and MF methods being imported from default statistics libraries. The DAE's architecture was taken from the MIDA paper [7], the VAE's from the MIWAE paper [9] and the GAN's from the GAIN paper [18]. Each method's specific parameters were determined through grid search performed on the datasets reserved for testing.

The three generative methods (MIDA, MIWAE and GAIN) were modified to include an early stopping criterion. Each training process utilizes 95% of the training data provided for actual training, saving the remaining 5% as a validation set. With each training epoch, the model attempts to generate values for the validation set, after which the respective RMSE is computed and stored. The training process reaches an early stop if 30 epochs elapse without the validation error decreasing.

We selected ten training datasets that range from sets high numbers of observations and attributes to sets with very few observations. Notably, we selected datasets of greater, smaller and similar sizes to the Alzheimer's Disease prediction datasets, on which this paper focuses. All these datasets are complete, with the majority containing a mix of continuous and categorical data. They contain real data and were obtained from public online platforms. The names, initialisms and dimensions of the 10 datasets utilized are present in Table 2

Table 2: Datasets utilized for testing with their respective number of observations and attributes. Datasets initialisms are displayed in parenthesis, for future references

Datasets	Obs	Att
World Happiness Report (WH)	156	10
Heart Disease UCI (HD)	303	14
Boston Housing (BH)	506	14
Pima Indians Diabetes (PI)	768	9
Breast Cancer Wisconsin (BC)	569	30
Red Wine Quality (RW)	1599	12
Gender Recognition by Voice (GR)	3168	20
Predicting a Pulsar Star (PS)	17898	8
House Sales in King County (KC)	21613	17
UFC-Fight Historical Data (UF)	3582	158

3.2. Classification

Before beginning the experimental work, we determined a pipeline setup. The setup needed to be versatile and allow for swapping methods at any point along the classification pipeline. Assuring this allowed us to compare different techniques at each step, while maintaining an otherwise identical setup. To determine the final pipeline's configuration we would need to perform several tests on the different state of the art technologies explored previously. Figure 2 illustrates the proposed classification pipeline.

The first step in the pipeline is to split the dataset into a Training dataset and a Validation dataset, where the goal is to correctly classify the validation set. After the split, both datasets go through a data cleaning step, where attributes with more than 70% missing data are eliminated.

This setup utilizes 5-Fold Cross Validation with fold randomization which is repeated 10 times. The remainder of the data pre-processing steps are performed within each iteration of the cross validation process. After the train/test division in the CV process, the selected MVI and FS methods are performed consecutively on the training dataset. The trained MVI methods are then used to impute the testing and validation datasets based on the training data. The features determined by the FS methods are selected from the training and validation sets as well. The order in which these two steps are performed is not a consensual topic within the scientific community. In T. Pereira's PhD dissertation [6], FS was performed before MVI. Considering that a substantial part of our work was performed on the MVI step, we opted to perform it first so as to allow the MVI methods to potentially highlight the importance of some features.

Afterwards, class balancing is performed exclusively on the training dataset ahead of the learning process, completing the data pre-processing steps. The selected classifier is used to learn the training data and then classify both the test and validation sets. To assess the results, one or more evaluation metrics are selected and used to compare the predicted values to the expected ones. The final results presented are averages of all the results obtained during the CV process.

The methods selected to be tested on this pipeline were the ones presented in section 2, with the addition of a specific set of features to the FS step. This set of features will be referred as "T. Pereira's FS" (or simply TP), and it consists of the best features chosen for classification in Telma Pereira's PhD dissertation [6]. Each model's specific parameters were determined through grid search performed on the CCC-3 datasets.

4. Results & discussion

4.1. Missing Value Imputation

In order to evaluate the performance of the different methods for missing value imputation previously explored, we opted to setup three different experiments. These experiments were structured as follows:

4.1.1 Influence of the Initial Imputation

All the generative models were described in their respective papers as utilizing Zero-Imputation. Meaning that, during the imputation process, all missing values are initially replaced by zero. With this information, we saw an opportunity to test whether using Mean-Imputation initially would provide better results.

To test this hypothesis, the three generative methods were utilized as well as the ten testing datasets. For each method-dataset pairing, eight measurements were performed, with the amount of missing data in the range of 10% to 45%, with intervals of 5%. In each of these measurements, both Zero-Imputation and Mean-Imputation were performed, and the RMSE value for each imputation was stored. For each of these measurements 5-Fold Cross-Validation was used.

The results obtained allowed us to withdraw the following conclusions:

- Dataset Size - The size of the dataset did not appear to have any major influence on the results observed. Even so, it is possible to note that, as the dataset's dimensions increase, the difference in the error obtained from using either of the methods decreases.

- Missing Rate - The Zero-Imputation method's results, when compared to the Mean-imputation ones, improved as the missing rate increased. This means that, in test cases where the Mean-Imputation method provided overall better results, the difference in error

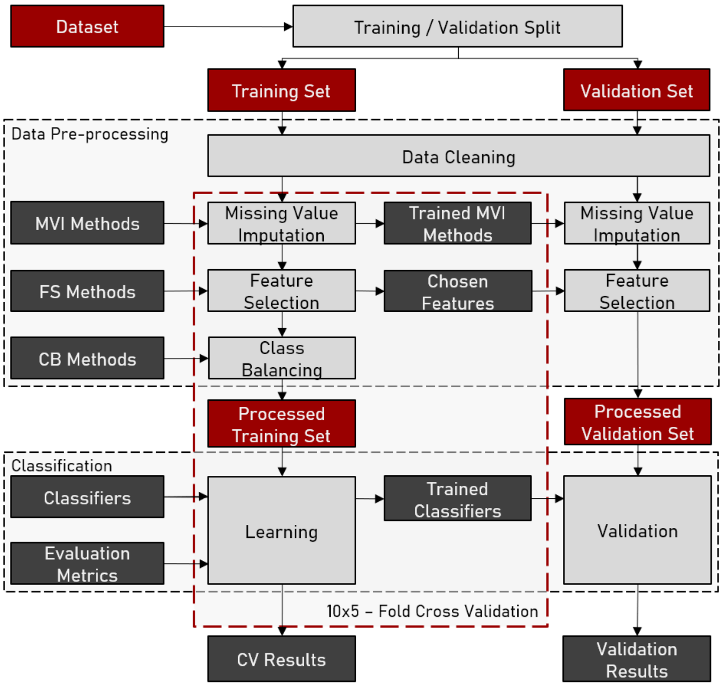

Figure 2: Pipeline for the entire classification process, including the evaluation of the obtained results

would decrease with the increase of missing data. Or in the case that both methods performed comparably overall, the Zero-Imputation method would have the better results for higher values of missing data.

Overall, the difference in results was never overpowering, nevertheless both the MIWAE and MIDA methods performed better with the use of Mean-Imputation,on average. The GAIN method, on the other hand, performed identically with either of the initial imputation methods. Considering this, from this test onward, the MIWAE and MIDA methods will utilize Mean-Imputation while the GAIN method will utilize Zero-Imputation.

4.1.2 Influence of Dataset Size

In the second set of experiments we aimed to understand how the dimensions of the dataset being used can influence the MVI methods explored. To measure that we decided to pick a single dataset and divide it into progressively bigger subsets, for each experiment. The datasets with the highest amount of observations (House Sales in King County) and the highest amount of attributes (UFC-Fight Historical Data) were chosen for experiments varying the amount of observations and attributes respectively. The subsets were created according to an exponential scale. The first test had subsets created from the KC dataset with the amount of observations equal to $2^n, n \in [7, 14]$. The second test had subsets created from the UF dataset with the amount of attributes equal to $2^n, n \in [3, 7]$. We evaluated the three generative methods described previously (MIWAE, MIDA and GAIN), as well as the state-of-the-art discriminative methods (MICE and MF) and also OSM for reference.

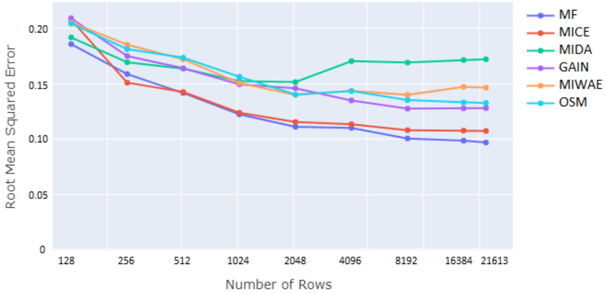

Figure 3: RMSE for imputation performed by all the MVI methods (MF, MICE, MIDA, GAIN, MIWAE, OSM) on the House Sales in King County Dataset (KC) with 20% missing data, for different numbers of observations.

First, we test how the length of the dataset can affect the imputation process. In figure 3 we can see the results for the imputation performed on the exponentially increasing subsets of the KC dataset. As was expected, an increase in the amount of observations results in a decrease in RMSE for all of the MVI methods. The increase in the number of observations affected all the MVI methods similarly, with the exception of the MIDA and MIWAE methods. These two methods struggled with the increase in the dataset's size, with their error values eventually stagnating, while the others kept decreasing. In this experiment the MICE and MF methods still performed the best out of all.

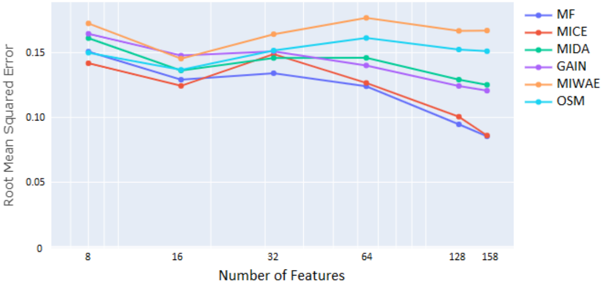

Figure 4: RMSE for imputation performed by all the MVI methods (MF, MICE, MIDA, GAIN, MIWAE, OSM) on the UFC-Fight Historical Data Dataset (UF) dataset with 20% missing data, for different numbers of attributes.

Figure 4 presents the results for the imputation performed on the subsets of the UF dataset with an exponentially increasing number of features. We can note that the OSM method used for reference, resulted in fairly consistent RMSE values, which was to be expected. Again, as was expected, an increase in the amount of features resulted in a slight decrease in RMSE for all of the MVI methods, with the exception of the MIWAE method which returned somewhat constant values. The increase in the number of features affected all other MVI methods in a comparable manner. Similarly to the previous test, in this one the MICE and MF methods still performed the best out of all methods, with the MIDA and GAIN presenting promising results.

4.1.3 Comparison of MVI methods

In the final set of experiments we aimed to compare the overall performance of all MVI methods explored so far, in order to determine which models perform the best. To perform this experiment, all the imputation methods previously described were used, on the ten datasets. For each imputation method-dataset pairing, eight measurements were performed, with the amount of missing data $\in [0.1, 0.45]$. For each of these measurements 5-Fold Cross-Validation was used.

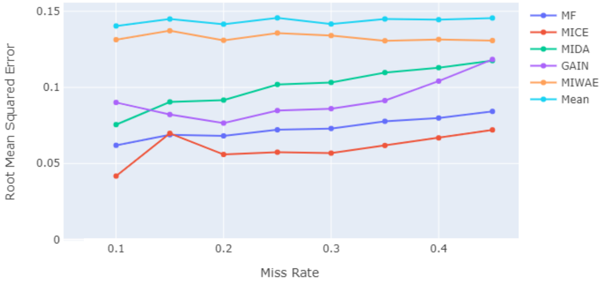

Figure 5: RMSE for imputation performed by all the MVI methods (MF, MICE, MIDA, GAIN, MIWAE, OSM) on the Breast Cancer Wisconsin Dataset (BC) for values of missing data $\in [0.1, 0.45]$.

In figure 5 we can see the RMSE for each of the MVI methods on the BC dataset. Out of the 10 datasets utilized, the BC dataset is the most similar in size to the CCC-3 datasets explored in this paper. From the obtained results we can note the following conclusions:

- Dataset Size - The MICE and MF methods were the clear best performers in the larger datasets, confirming what we observed previously. The MIDA method had the worst drop in performance as the datasets increased in size. Although it was one of the best performers in datasets of smaller size, along with MICE and MF. The GAIN and MIWAE methods had rather consistent results, independently of the amount of observations in the dataset utilized. MIWAE had worse results in datasets with more features.

- Missing Rate - Compared to the remaining methods, MF, MIDA and MIWAE presented extremely consistent results as the rate of missing data increased. On the other hand, the GAIN and MICE methods sometimes struggled as the amount of missing data neared 50%. Overall, the MF, MICE and MIDA methods performed the best, regardless of missing rate. GAIN performed better than MIWAE for smaller missing rates, and the reverse was true for larger missing rates.

Overall, MF performed better than the remaining methods in the majority of the experimental setups. Immediately following it, the MICE and MIDA methods were consistently among the best methods. The GAIN method had situational setups that allowed it to outperform some of the other methods. It presented particularly good results in larger datasets with smaller miss rates. Inversely, MIWAE's best performing test scenarios were those with smaller datasets and larger amounts of missing data.

4.2. Classification

The goal in this section was to determine the combination of methods that would allow us to achieve the best classification results possible for each dataset. Considering the amount of variables at play, our tests were performed according to a funneling strategy, where some of those variables are eliminated with each test that is performed.

4.2.1 Previous Result Recreation

First, since this paper aims to build upon the work performed on Telma Pereira's PhD dissertation [6], we decided to replicate T. Pereira's pipeline configuration to use as a baseline for our results. The pipeline was set up using the SMOTE method for handling class imbalance, the OSM method for missing value imputation and seven different classifiers: NB, DT, SVM RBF, SVM Poly, KNN, RF and LR.

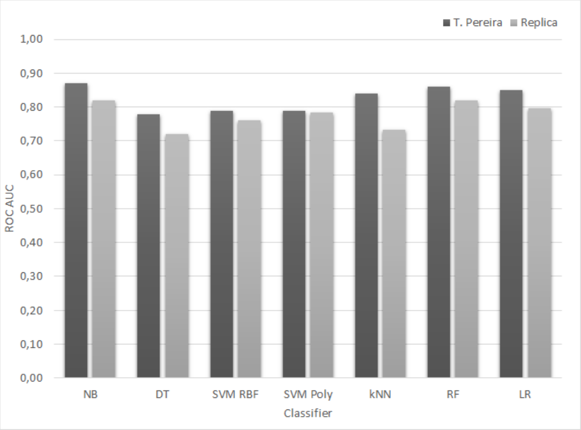

Figure 6: Comparison of the ROC AUC score between results presented on Telma Pereira's PhD dissertation (dark grey) and results obtained for this thesis (light grey), while using comparable pipeline setups, for the 5Y dataset.

The results obtained with our replicated setup lagged behind those of T.Pereira, albeit not by an insurmountable amount. Regrettably, the datasets seem to have suffered some reorganization, which means that not all the features utilized in T. Pereira's dissertation seem to be present in the latest version of the datasets. The most notable differences come when using the DT and KNN classifiers, which consistently presented worse results for all datasets. On the other hand, the SVM Poly classifier returned identical results to the ones expected. Overall, the NB, RF and LR classifiers, which were consistently the best performers in Pereira's work, present some of the most similar results. This similarity allows us to have some confidence in emulated pipeline setup.

4.2.2 Determining the best Classifiers

In the first set of experiments our aim was to reduce the list of potential classifiers available. The tests were performed on the eight classifiers described in section 2, after grid search was performed to tune their respective parameters. All tests were performed exclusively on the CCC-3 Lisbon datasets, utilizing 10×5-Fold Cross Validation, with an 80/20 train/validation split.

The features in T. Pereira's FS were utilized and the SMOTE method was picked to address class imbalance. These methods were chosen according to the previous replica. For MVI we opted to use the MissForest method, as we concluded in section 4.1 that it was the one that provided us with the best results.

The classification scores obtained on the 4Y testing dataset are available in table 3. From the obtained results some patterns can be noted, such as the Naive Bayes classifier consistently returning better Accuracy and ROC AUC results than the remaining classifiers. Nonetheless, this test was not aimed at selecting the single best classifier, as other variables that were not yet tested can have direct impact on the classifier's performance. Therefore, four classifiers were selected as the best performers: NB, SVM Poly, RF and NN.

Table 3: Classification results of 10×5-Fold Cross Validation on the 4Y dataset.

Classifier	Accuracy	AUC
NB	0,81 ± 0,03	0,81 ± 0,03
DT	0,65 ± 0,04	0,65 ± 0,05
SVM RBF	0,73 ± 0,06	0,73 ± 0,06
SVM Poly	0,76 ± 0,04	0,75 ± 0,04
kNN	0,69 ± 0,05	0,68 ± 0,05
RF	0,77 ± 0,03	0,77 ± 0,03
LR	0,75 ± 0,05	0,75 ± 0,05
NN	0,75 ± 0,04	0,75 ± 0,04

4.2.3 Determining the best method for Class Balancing

Having reduced the number of classifiers, the next step is to do the same with the proposed methods for class balancing. To do so, the same classification setup was utilized for the five class balancing methods being tested.

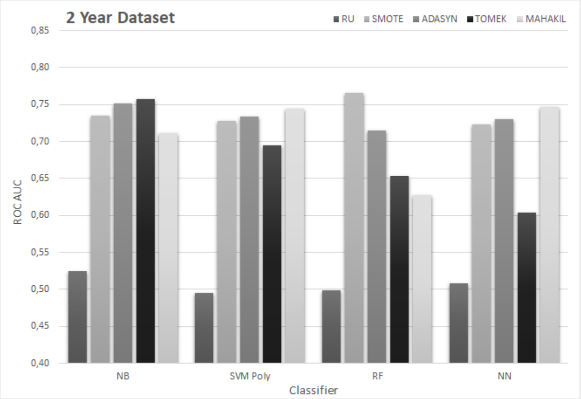

Figure 7: Comparison of the ROC AUC score obtained while using five different methods for class balancing (respectively Random Undersampling, SMOTE, ADASYN, TOMEK Links and MAHAKIL). The results were obtained using the four best performing classifiers mentioned previously on the 2Y dataset.

The ROC AUC scores, grouped according to the classifiers that were utilized, are available in figure 7. The obtained results allow for some immediate observations:

- The Random Undersampling method is clearly inefficient and significantly worse than others.
- Even though the TOMEK Links method was overall one of the best, the Neural Network

classifier performs worse on average when being paired with this method.

- The overall best ROC AUC scores for each dataset were obtained with the SMOTE method (for datasets 2Y, 3Y and 4Y) and with the TOMEK Links method (for the 5Y dataset).

In conclusion, barring the RU method, all other methods result in fairly comparable scores. Moving forward we decided to reduce the list of Class Balancing methods to SMOTE and TOMEK Links, for the following reasons: On average, both of these were among the three best performing methods. Furthermore, since the TOMEK Links method performs undersampling, choosing it allows for more diversity in future testing scenarios.

4.2.4 Determining the best Missing Value Imputation method

In this set of experiments, our aim is to apply some of the research performed regarding MVI. A similar pipeline setup to the ones utilized previously was selected. The MIWAE method was not tested due to it being the worst performer in the last section for similar datasets.

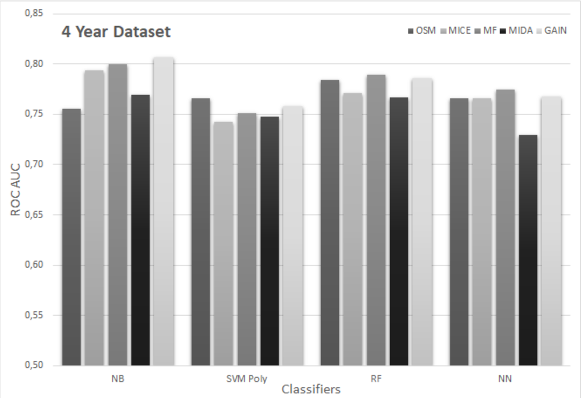

Figure 8: Comparison of the ROC AUC score obtained while using five different methods for MVI (respectively Overall Sample Mean, MICE, MissForest, MIDA and GAIN). The results were obtained using the four best performing classifiers (NB, SVM Poly, RF and NN) and the SMOTE method for class balancing on the 4Y dataset.

Some of the ROC AUC scores, grouped according to the classifiers that were utilized, are available in figure 8. The obtained results allow for some immediate observations:

- The MIDA method was the worst performer, seemingly struggling more with the more unbalanced datasets (2Y and 3Y).

- The MICE method performed significantly better when paired with the NB classifier than with any other classifier.

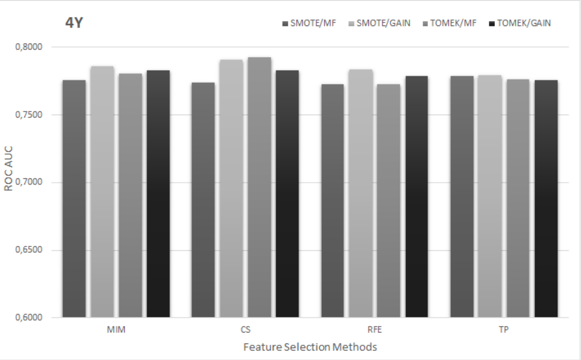

Figure 9: Comparison of the ROC AUC score obtained while using four different methods for FS (respectively MIM, CS, SVM RFE and the features used in T. Pereira's dissertation). Each FS method was tested in four setups that result of a combination of two CB methods (SMOTE and TOMEK) and two MVI methods (MF and GAIN). Each column contains the average of the scores obtained with the four best performing classifiers (NB, SVM Poly, RF and NN) on the 4Y dataset.

- The RF classifier had the most consistent results, with relatively small variations for the different MVI methods.

Further analysing the results, we can note that the class balancing method seemed to have minimal impact on the performance of the different MVI methods. Overall setups with SMOTE method returned better results than ones with TOMEK, and the best ROC AUC result for each dataset was always obtained utilizing SMOTE.

Overall, the two best performing MVI methods were MF and GAIN. Not only did these contribute to some of the best results for each dataset, but they were also consistently among the best methods regardless of the classifier being used. Since there is a need to further restrict the amount of variables being used for testing, we chose to use those two methods in the following testing scenarios.

4.2.5 Determining the best method for Feature Selection

Finally, in this section, several methods for Feature Selection are compared. Again, the pipeline follows a similar setup to the previous ones. Considering all the variables being tested, this resulted in a total of 64 setups to compare the performance of the FS methods. With this amount of information, we redirected our goals to group some of the results in order to find correlations or patterns, as well as having as many variables as possible so as to find the best setup configuration for each of the datasets.

Figure 9 contains a graph with the results for the 4Y dataset. In order to ease understanding, each column contains the average of the scores obtained with the four different classifiers (NB, SVM

Poly, RF and NN) for that exact setup. While it may ease the illustration of the results, grouping the data in such a manner can “dilute” the results, occluding some important outlier cases. Knowing this, all the following conclusions were drawn from the original data collected. Some of the conclusions noted, organized by the different variables tested, were:

- Classifier - The NB classifier was consistently worse when paired with the SVM RFE method. The same relation was noticeable between the NN classifier and the set of features used in T. Pereira’s work (which can be relevant seeing as NNs were not utilized in T. Pereira’s work).
- MVI Methods - GAIN was significantly more consistent than MF, and setups with GAIN presented smaller variability when swapping between datasets. The different FS methods did not seem to have a big impact on the performance of the MVI methods, with the notable exception being CS which returned significantly better results when paired with GAIN.
- CB Methods - Generally the SMOTE method was the ideal choice for the more unbalanced datasets (2Y and 3Y), while TOMEK performed better with the more balanced ones (4Y and 5Y). SMOTE performed consistently regardless of the FS method. TOMEK always performed better when paired with CS, and worse when paired with SVM.

With every experiment performed so far, the difference in scores tends to be smaller and less noticeable, and that is evident in this set of measurements. Overall the four different techniques for FS performed comparably to each other, with only the SVM RFE performing slightly worse than all other methods.

While there was no clear best performing technique for FS, it was evident that some combinations of different parameters work significantly better than others, as was noted before. It is also relevant to point out the impact that each of these techniques had on the size of the datasets being tested. On average, the CS method resulted in datasets with the highest amount of features, while the features from T. Pereira were the smallest group (always 42 features).

4.2.6 Selecting the best Setup

The last set of experiments allowed us to test 64 different setups for each of the four datasets. Having all these measurements allows us to finally compare complete setups and determine the best ones. Table 4 contains the configurations of the setups that resulted in the highest ROC AUC scores for each of the datasets.

Dataset	Classifier	CB	MVI	FS
T. Pereira	NB	SMOTE	OSM	TP
2Y	NN	SMOTE	GAIN	CS
3Y	NB	SMOTE	MF	MIM
4Y	RF	TOMEK	MF	CS
5Y	NB	TOMEK	GAIN	CS

It is also relevant to point out that the best setup for all the datasets on average is the setup that returned the best results for the 5Y dataset (utilizing NB, TOMEK, GAIN and CS). In table 5 it is possible to compare the original results to the ones obtained with these setups. The original results were obtained in section 4.2.2 with the replica of T. Pereira’s setup (utilizing NB, SMOTE and OSM). These can be compared with our two proposed setups: The first proposal being the best individual setup for each of the years, *i.e.*, the setups described in tab:BestSetups for each of the datasets; The second proposal being the best overall setup for all the datasets (NB, TOMEK, GAIN and CS).

Table 5: Comparison between the original results obtained in section 4.2.2 and results obtained with two different proposed setups (best individual setup for each year and best overall setup, respectively).

Dataset	Setup	ROC AUC	Accuracy
2Y	Original	0.73	0.75
	Proposal #1	0.79	0.78
	Proposal #2	0.75	0.76
3Y	Original	0.77	0.76
	Proposal #1	0.79	0.78
	Proposal #2	0.79	0.78
4Y	Original	0.76	0.72
	Proposal #1	0.81	0.81
	Proposal #2	0.79	0.79
5Y	Original	0.76	0.71
	Proposal #1	0.82	0.81
	Proposal #2	0.82	0.81

Finally we decided to return to the testing scenarios utilized in T. Pereira’s work, described in section 4.2.1. In these scenarios the entire Lisbon CCC-3 dataset is utilized for training, while the Coimbra CCC-3 dataset is used as validation. Table 6 and table 7 contain the Cross-Validation results obtained from the Lisbon dataset, and the validation results obtained from the Coimbra dataset, respectively. In the Lisbon CV results (table 6) we can note a small improvement from the results obtained with replicated setup, although even the improved results lag behind those obtained in T. Pereira’s dissertation. Nonetheless, as there is no way for us to recreate an exact replica of that setup, T. Pereira's results can only be used as a reference and not as direct comparison.

In the Coimbra results (table 7) however, the results obtained from the proposed setups were equal or worse than those obtained with the replica setup. This discrepancy in results could suggest that our proposed setups are over-fit to the training dataset, but the fact that all the measurements were obtained using Cross Validation and an additional Validation set eliminates some of those concerns. Another possibility could be that the methods explored in this thesis managed to bring out more information from the features that are present in the Lisbon dataset but are entirely missing from the Coimbra dataset.

Table 6: Comparison between the results presented on Telma Pereira's PhD dissertation (T. Pereira), the results obtained while replicating that setup (Replica), and results obtained from the two best setups proposed in this thesis (Proposal #1 and Proposal #2). These scores were obtained from the CV process performed on the CCC-3 Lisbon dataset.

	AUC			
	2Y	3Y	4Y	5Y
T. Pereira	0.83	0.85	0.86	0.87
Replica	0.77	0.80	0.81	0.82
Proposal #1	0.78	0.80	0.82	0.84
Proposal #2	0.78	0.80	0.81	0.83

Table 7: Comparison between the results presented on Telma Pereira's PhD dissertation (T. Pereira), the results obtained while replicating that setup (Replica), and results obtained from the two best setups proposed in this thesis (Proposal #1 and Proposal #2). These scores were obtained from the validation dataset (CCC-3 Coimbra).

	AUC			
	2Y	3Y	4Y	5Y
T. Pereira	0.66	0.67	0.64	0.63
Replica	0.68	0.66	0.61	0.59
Proposal #1	0.63	0.66	0.61	0.53
Proposal #2	0.65	0.62	0.60	0.53

5. Conclusions

In this work we studied the prognosis and evolution of Alzheimer's Disease in patients initially diagnosed with Mild Cognitive Impairment. Being able to correctly diagnose AD at early stages and being able to predict its evolution can have significant social and economic impacts globally. To tackle this issue, we utilize data collected from Neuropsychological tests. This data comes with some downsides, such as incongruities based on the collection sites and responsible teams, small sample sizes and unbalanced datasets. Our approach to overcome these problems was to research and implement state-of-the-art techniques and algorithms that address areas like Missing Value Imputation, Class Balancing, Feature Selection and Classification.

Initially we implemented six different algorithms to address MVI in datasets. In order to test those algorithms, we collected 10 datasets of diverse real-world data and set up a series of experiments. Each of those experiments allowed us to compare the different algorithms and draw conclusions based on their performance.

Afterwards we constructed a modular data mining pipeline, capable of swapping between different methods at each stage. Before performing any tests, we had implemented a total of eight classifiers, five CB methods, six MVI methods and four FS techniques. In order to establish a baseline, we attempted to replicate the pipeline setup present in T. Pereira's work [6]. After that, we recorded Accuracy, Sensitivity, Specificity and ROC AUC scores obtained on two different configurations of the datasets: The first where only the Lisboa CCC-3 dataset was used for both training and validation, and the second where the Lisboa CCC-3 set was used for training and the Coimbra CCC-3 for validation.

Having acquired baseline scores, we could now attempt to determine the best pipeline setup. To do so we adopted an incremental testing approach where, at each stage, all but one steps in the pipeline would be fixed so that the remaining step could be tested with the different methods available for it. After each testing iteration the worst performing methods would be discarded. By the final set of tests, the available methods had been restricted to four classifiers, two CB methods, two MVI methods and four FS techniques. Pipelines with every possible combination of these methods were tested, allowing us to choose the combinations that resulted in the highest classification scores. From those scores we proposed two solutions to the original problem: The first proposal consists of the best setup for each individual dataset; the second proposal consists of the setup that resulted in the highest score on average for all the datasets.

Both proposed setups ended up returning equal or higher scores than the established baseline. This is true for the validation scores in the first dataset configuration (only using the Lisbon CCC-3 dataset) as well as the cross-validation testing scores in the second dataset configuration (using Lisbon and Coimbra CCC-3 datasets). However, the proposed setups were not able to overcome the baseline validation scores (Coimbra CCC-3 dataset) of the second configuration.

References

[1] A. Association. 2019 Alzheimer ' s disease facts and figures. *Alzheimer's & Dementia*, 15(3):321–387, 2019.

[2] K. E. Bennin, J. Keung, P. Phannachitta, A. Monden, and S. Mensah. Mahakil: Diversity based oversampling approach to alleviate the class imbalance issue in software defect prediction. *IEEE Transactions on Software Engineering*, 44(6):534–550, 2018.

[3] A.-M. Bidgoli and M. N. Parsa. A hybrid feature selection by resampling, chi squared and consistency evaluation techniques. *World Academy of Science, Engineering and Technology*, 68:276–285, 2012.

[4] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[5] D. Devi, B. Purkayastha, et al. Redundancy-driven modified tomek-link based undersampling: a solution to class imbalance. *Pattern Recognition Letters*, 93:3–12, 2017.

[6] T. Filipa, L. De, M. Pereira, S. Alexandra, and C. Madeira. Prognostic models targeting time to conversion, stable predictors, and reliability at patient-level: Predicting progression from mild cognitive impairment to dementia Biomedical Engineering. (February), 2019.

[7] L. Gondara and K. Wang. Mida: Multiple imputation using denoising autoencoders, 2017.

[8] Y. E. Kurniawati, A. E. Permanasari, and S. Fauziati. Adaptive synthetic-nominal (adasyn-n) and adaptive synthetic-knn (adasyn-knn) for multiclass imbalance learning on laboratory test data. In *2018 4th International Conference on Science and Technology (ICST)*, pages 1–6, Aug 2018.

[9] P.-A. Mattei and J. Frellsen. Miwae: Deep generative modelling and imputation of incomplete data. *arXiv preprint arXiv:1812.02633*, 2018.

[10] R. Nallapati. Discriminative models for information retrieval. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 64–71, 2004.

[11] T. Pereira, L. Lemos, S. Cardoso, D. Silva, A. Rodrigues, I. Santana, A. De Mendonça, M. Guerreiro, and S. C. Madeira. Predicting progression of mild cognitive impairment to dementia using neuropsychological data: A supervised learning approach using time windows. *BMC Medical Informatics and Decision Making*, 17(1):1–15, 2017.

[12] P. M. Prince, G.-c. Ali, and G.-c. Ali. World Alzheimer Report 2015 The Global Impact of Dementia. 2015.

[13] P. Royston, I. R. White, et al. Multiple imputation by chained equations (mice): implementation in stata. *J Stat Softw*, 45(4):1–20, 2011.

[14] D. Silva, M. Guerreiro, J. Maroco, I. Santana, A. Rodrigues, J. B. Marques, and A. de Mendonça. Comparison of four verbal memory tests for the diagnosis and predictive value of mild cognitive impairment. *Dementia and geriatric cognitive disorders extra*, 2(1):120–131, 2012.

[15] D. J. Stekhoven and P. Bühlmann. MissForest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 10 2011.

[16] P. Viola and W. M. Wells III. Alignment by maximization of mutual information. *International journal of computer vision*, 24(2):137–154, 1997.

[17] K. Yan and D. Zhang. Feature selection and analysis on correlated gas sensor data with recursive feature elimination. *Sensors and Actuators B: Chemical*, 212:353–363, 2015.

[18] J. Yoon, J. Jordon, and M. van der Schaar. Gain: Missing data imputation using generative adversarial nets, 2018.