# Hierarchical Modelling for ALS Prognosis: Predicting the Progression Towards Critical Events

Notebook for the iDPP Lab on Intelligent Disease Progression Prediction at CLEF 2022

Ruben Branco<sup>\*,†</sup>, Diogo F. Soares<sup>\*,†</sup>, Andreia S. Martins<sup>†</sup>, Eleonora Auletta, Eduardo N. Castanho, Susana Nunes, Filipa Serrano, Rita T. Sousa, Catia Pesquita, Sara C. Madeira and Helena Aidos

LASIGE, Faculdade de Ciências, Universidade de Lisboa, Portugal

#### Abstract

Amyotrophic Lateral Sclerosis is a neurodegenerative disease that leads to a patient's progressive loss of cognitive and motor capacities. Its mechanisms are poorly understood, meaning no cure is available. Prognosis based on clinical data thus becomes fundamental to make treatment plans; however, the heterogeneous progression amongst patients presents a considerable challenge and motivates precision medicine research.

iDPP :: CLEF 2022 aims to develop novel methodologies for predicting ALS disease progression in the form of a challenge, enabling the research community to combine efforts to improve current methods. This paper reports on our participation in task 2 to predict critical events (NIV, PEG and DEATH) and the expected time window for them to occur.

The proposed approach in this work combines a methodology based on pattern mining, which uncovers relevant patterns in clinical data, and state-of-the-art machine learning methodologies. We design a hierarchical strategy in which the top layer predicts the event, and the bottom layer the time window specialized on each possible event.

Results show that our event prediction layer is effective in predicting the event most likely to occur in a given patient, however, the more significantly challenging task of predicting the time window to occurrence yielded a subpar performance, calling for future research.

#### Keywords

Amyotrophic Lateral Sclerosis, Prognostic Prediction, Disease Progression Patterns

# 1. Introduction

Amyotrophic Lateral Sclerosis (ALS) is a chronic disease affecting the neurons of the human motor system. It is characterized by progressive loss of motor functions, such as walking, speaking and eating. In later stages of the disease, breathing difficulties also arise, with respiratory

risousa@ciencias.ulisboa.pt (R. T. Sousa); clpesquita@ciencias.ulisboa.pt (C. Pesquita); sacmadeira@ciencias.ulisboa.pt (S. C. Madeira); haidos@ciencias.ulisboa.pt (H. Aidos)

CLEF 2022: Conference and Labs of the Evaluation Forum, September 5-8, 2022, Bologna, Italy

<sup>\*</sup>Corresponding author.

<sup>&</sup>lt;sup>†</sup>These authors contributed equally.

<sup>☆</sup> rmbranco@ciencias.ulisboa.pt (R. Branco); dfsoares@ciencias.ulisboa.pt (D. F. Soares);

amartins@lasige.di.fc.ul.pt (A. S. Martins); eauletta@ciencias.ulisboa.pt (E. Auletta); ejcastanho@ciencias.ulisboa.pt (E. N. Castanho); scnunes@ciencias.ulisboa.pt (S. Nunes); fserrano@lasige.di.fc.ul.pt (F. Serrano);

Sachiaden a Weten class. Ints boa. pt (5, C. Maden a), flatto Sweeten class. Ints boa. pt (11.
)
2022 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

failure being the main cause of death in ALS patients [1].

With no known cure for ALS, current treatments are centered on both retarding the natural progression of the disease and improving patients' quality of life. Common therapies include permanent ventilation and enteral nutrition, especially at later stages of the disease [2].

However, the mostly heterogeneous nature of ALS, in aspects such as affected areas (both at onset and throughout the disease course), as well as progression and survival times, makes it challenging for clinicians to effectively diagnose and anticipate the evolution of ALS.

Thus, machine learning techniques have been used to aid in unraveling disease mechanisms and clinical decision making [3]. Such methods have been applied to the identification of key biological markers for both diagnosis and prognosis [3], prediction of survival and end-stage events [4], as well as discovery of patient groups with similar disease progression courses [5] or treatment response [6].

The iDPP :: CLEF 2022 challenge<sup>1</sup> uses highly curated data from real ALS patients, followed at clinical institutions in Lisbon, Portugal, and Turin, Italy. The focus of the challenge is prognostic prediction in ALS. Particularly, task 2 focuses on the prediction of end-stage events: initiation of non-invasive ventilation (NIV), percutaneous endoscopic gastrostomy (PEG) and death, as well as their respective times to occurrence. These subjects are of high importance in ALS, given the fast-progressing nature of the disease and how crucial timely administration of ventilation and enteral nutrition can be in improving patients' life expectancy.

In our contribution to iDPP  $\therefore$  CLEF 2022, we propose a hierarchical approach, with a first-stage event prediction, followed by specialized models predicting the time window to a particular event. Our procedure is three-fold: first, it implements the innovative approach proposed by Carreiro et al. [7] to create patient snapshots based on clustering with constraints, thus organizing patient records in an efficient manner. This methodology was observed to improve the state-of-the-art classifiers by providing a more consistent description of the patients condition. Second, it uses the pattern-based approach described by Martins et al. [8] that incorporates recent advances on temporal pattern mining to the context of classification. This approach performs end-stage event predictions from the previous step we learned specialized models using the original features to predict the time window to that event. This two-stage prediction approach aims to promote homogeneity and lessen the impact of class imbalance, in comparison to performing one single multilabel task.

The rest of the paper is organized as follows: Section 2 presents the related work; Section 3 explains the applied hierarchical approach for task 2 predictions; Section 4 describes the setup done for the experiments; Section 5 discusses the core results ; and finally, Section 6 concludes the paper and gives some hints about future work.

# 2. Related Work

The non-appearance of a curative treatment leads to the need for researching new care treatments that prolong survival and improve the quality of ALS patients' life. Patients with ALS and their family benefit from regular patient follow-up in anticipating and preparing for death

<sup>&</sup>lt;sup>1</sup>https://brainteaser.health/open-evaluation-challenges/idpp-2022/

[9]. Because of their confusing and dynamic illness process, people with ALS have a few interdisciplinary demands. They benefit from individualised rehabilitation programs to improve independence, function, and safety. These techniques can also help with symptom management and quality of life. [10].

In the last years some personalized methods based on machine learning were proposed for the ALS prognosis. Carreiro et al. [7] was pioneer in proposing prognostic models to predict the need for non-invasive ventilation support based on clinically defined time windows. More recently, Pires et al. [11] stratified patients according to their state of disease progression achieving three groups of progressors (slow, neutral and fast), and proposed specialized learning models according to this groups with the same target prediction. They further used patient and clinical profiles with promising results [12].

Regarding survival prediction, some relevant studies were conducted using machine learning techniques. Tavazzi et al. [13] used 3 months of static and dynamic data to predict if a patient will succumb to the disease 3 years from its first assessment. Grollemund et al. [14] designed a probabilistic method to predict short-term survival (1 year) in ALS patients.

Predicting the progression of the functional domains (ten questions) assessed by the wellknown functional scale, the ALSFRS-R, was also investigated by Gordon and Lerner [15]. They modeled a multiclass classifier using demographic, respiratory assessments, genetic data, and other dynamic data to predict the values of each ALSFRS-R question at the time of the last patient visit.

Martins et al. [8] recently proposed combining itemset mining with sequential pattern mining to uncover disease presentation and progression patterns in ALS patients and utilize these patterns to forecast the need for NIV. In a similar approach with the same prognostic target, Matos et al. [16] suggested a classifier based on biclustering. Biclustering was used to locate groups of patients with similar values in subsets of clinical features (biclusters), which were then combined with static data as features. Apart from being promising, none of these methods took into account the characteristics' temporal relationship. A preliminary assessment of the role of conventional triclustering techniques in predicting breathing support needs in ALS was provided by Soares et al. [17].

# 3. Methodology

The strategy for our contribution to the iDPP :: CLEF 2022 is based on a hierarchical approach, which we named Hierarchical Prognosis. The strategy comprises two stages: the event and time to event prediction. Due to its hierarchical nature, the predictions made at the upper level, the event layer, are propagated downwards to the time to event prediction. This hierarchical approach benefits from (i) less heterogeneity amongst the population of patients used for predictions; (ii) reduced class imbalance within event data subsets, improving predictions for the less common (event, time window) pairs. The approach is broadly described in the diagram found in Figure 1.

This section covers the data preprocessing (Section 3.1), together with the functioning of the two levels of the hierarchical approach, the Event Prediction (Section 3.2) and Time to Event Prediction (Section 3.3) layers.



**Figure 1:** Hierarchical Prognosis for subtask 2A. In the upper layer, the classifier is trained with patient data to predict an event. For each event type, on the bottom layers, a classifier is trained on the corresponding subsection of the dataset to predict the time to event. A similar architecture is adopted for subtasks 2B and 2C.

## 3.1. Data Preprocessing

The data preprocessing stage encompassed two phases: (1) general processing of the raw datasets and (2) specific preprocessing concerning each predictive task, whether predicting the event or time to event.

Regarding the first step, we first preprocessed the dataset containing patients' information together with their clinical historical assessments of spirometry and the ALSFRS-R scale. Considering that spirometry and ALSFRS-R assessments were not done on the same day, we considered to merge these evaluations according to their performance dates, simplifying the learning instances dealing with temporal distribution and avoiding a large number of blank values within data. With this in mind, we followed the approach proposed by Carreiro et al. [7] to create patients snapshots which consist in grouping evaluations based on hierarchical clustering with constraints. In the original article the author used two constraints, given the predictive problem in hands. In this case we considered the creation of the snapshots stating only that the evaluations that compose a snapshot cannot belong to the same test as doctors do not prescribe the same test twice (we only merged spirometry tests with ALSFRS-R tests in this case).

The assessments were only grouped if they distance at least 3 months. The reference date for the resulting snapshot was defined as the median between the dates of its composing assessments. We coupled the corresponding static features and labels for each patient's snapshot. This process was done for the three subtasks 2a, 2b and 2c with their correspondent datasets.

The final datasets with the learning examples were composed by 3.728 snapshots for subtask 2a; 4.333 snapshots for subtask 2b; and 4.435 snapshots for subtask 2c.

Having these datasets created we moved to the second step of preprocessing, specific for each one of the predictive methods within our hierarchical approach.

## 3.2. Event Prediction

The first part of our strategy consists on predicting the occurrence of end-stage events, with each subtask focusing on a key event. Subtasks 2a and 2b are multilabel problems, each with 3 possible outcomes: NIV (2a), PEG (2b), DEATH and NONE. Subtask 2c is a binary problem, with DEATH and NONE as available events.

Prediction of event for task 2 was based on employing pattern mining algorithms from the SPMF library [18], and using occurrence of extracted patterns as features. While they might be less informative than the original variables, this approach allows for the entire clinical history to be taken into account without missing value imputation.

The procedure followed the approach described by Martins et al. [8], adjusting for the features and the multilabel problem at hand in subtasks 2a and 2b. After general preprocessing described in Section 3.1, the snapshot sets were firstly transformed into sequence databases for the pattern mining algorithms. Then, static and temporal patterns were retrieved, and finally new learning instances were built for classification.

Snapshot sets were firstly split by event type: {*NIV, DEATH, NONE*} for subtask 2a, {*PEG, DEATH, NONE*} for subtask 2b and {*DEATH, NONE*} for subtask 2c. The following preprocessing, up to and including pattern mining, is done independently for each event. We make this split in order to avoid extracting patterns only found in the most common event labels, which could impact event prediction. Then, static and temporal features were also split, as different pattern mining algorithms will be applied to both parts. Features were also selected at this stage, based on the percentage of non-missing values, which should be above the selected support threshold. Additionally, features with a large number of values (such as *occupation* and *limb onset type*) were also removed, as they could originate a prohibitively large amount of extracted patterns. ALSFRS-R individual questions were also not utilized due to this possibility.

Static and temporal feature sets were discretized, with pattern items being of the form *feature = interval* (if numeric) or *feature = value* (if categorical). Intervals were set as proposed by Martins et al. [8] and on value distributions. In the case of genetic features, where there were separate columns for Lisbon and Turin patients, the values were transformed so that single binary columns were kept. The resulting items were indexed as required by SPMF, with

sequence databases being written in the required format and each sequence representing one patient. In the static case, each sequence is comprised of the items in a single snapshot (since the information is repeated across all of the patient's snapshots). In the temporal sequences, a snapshot set is depicted in a single row, with temporal delimiters specifying the snapshots. For predictions made using only month 0 (M0) data, this transformation is solely applied to the set's first snapshot. Otherwise, we use all the available clinical history of the patients (6 month period – M6).

The sequence databases serve as input to the SPMF pattern mining algorithms, which were applied according to the followed approach. AprioriTID [19] and Fournier\_08 [20] were utilized to extract static and temporal closed patterns, respectively, using 25% as support threshold. Fournier\_08 has additional parameters, which were set as not to exclude any patterns based on overall length or temporal gaps.

The following step comprised on building the pattern sets to be added to the training data. As pattern mining was done separately for each event, there are two points to take into account: (1) there can be repeated patterns, and (2) it is unknown if patients in one event group verify the patterns of another event group. The first point is addressed by compiling a pattern list containing duplicate status, as well as available sequence identifiers and number of items and/or time points. At this stage, 1-item patterns are removed.

The available information in this list is used to build a binary matrix, with each row pertaining a patient and each column matching a singular pattern. Elements are 1 if the patient verifies the pattern, and 0 if otherwise or unknown. The unknown elements are dealt with through distance computation. Through the snapshot sets and pattern lists, a table containing all known feature values at each time point is computed for both patients and patterns. Discrete feature values, normalized between 0 and 1, are also added. These tables were named *patient* and *pattern vectors*.

In the case of static patterns, the euclidean distance between patient P and pattern Q is computed as:

$$d_t(P,Q) = \sqrt{\sum_{\tau \in \mathcal{T}} \sum_{f \in \mathcal{F}} (p_{t+\tau,f} - q_{\tau,f})^2},\tag{1}$$

where  $\mathcal{T}$  contains the pattern's time points and  $\mathcal{F}$  the set of features presented at each time point  $\tau$ . p and q take the normalized feature values for patient and pattern, respectively.

For temporal patterns, several distances must be computed, as the time stamps on the patterns are relative, and therefore a pattern can be found at various stages of the patient sequences. Thus, one distance is calculated for each potential starting point in the sequence.

However, when performing these calculations, we may encounter missing values on the patient end. In that case, the individual feature distance is set as the maximum possible distance to the value included in the pattern:

$$p_{t+\tau,f} = nan \to (p_{t+\tau,f} - q_{\tau,f}) = \max\{|0 - q_{\tau,f}|, |1 - q_{\tau,f}|\}$$
(2)

As in the proposed approach, this is only done if at least one value per pattern time point is known within the patient sequence. Otherwise, the distance is considered infinite. This results in a set of distances, out of which the shortest is selected. If a patient verifies the pattern in question, then at least one of the distances in the set will be 0. The corresponding similarity value, defined as:

$$s = \frac{1}{1 + \min d_t} \tag{3}$$

is then added to the training matrix. While this procedure allows for building both similarity or binary matrices (as the similarity value may only be added if it is 1), we have elected to only use the similarity matrix, as it provides more information than its binary counterpart.

Finally, the event class is added to the matrices. These datasets are comprised of 391, 492 patterns for subtask 2a; 276, 406 patterns for subtask 2b; 184, 298 patterns for subtask 2c, for M0 and M6, respectively.

Models for event prediction were trained using Random Forests with 200 classifiers. Sampling was performed through SMOTE [21], with a non-majority strategy (that is, only the majority event is not oversampled). Tests were done following a stratified train/test split, with the test set comprising 20% of the provided learning examples for each subtask. However, final predictions on the provided test set were made through Random Forest models trained on the complete training data.

In order to perform these predictions, however, the test set had to also be transformed into a similarity matrix. Thus, we applied a similar procedure to that used to compute the unknown similarity values on the training sets. Patient vectors were obtained for the patients in the test set and similarity values were computed between patient and pattern, using the same pattern vectors as in the training data.

## 3.3. Time to Event Prediction

In Section 3.2 we described the upper level of our hierarchical system, the event prediction layer. Given a new patient, the event most likely to happen in the near future is first predicted through this layer. The predicted event is then propagated to the lower level, the time to event layer, which is itself composed of different sub-layers depending on the event, as pictured in Figure 1.

The dataset is split by event type, creating different populations composed of patients with the same event type. For each event type, a classifier is trained to predict, based on the patients' features, one of the possible time windows (in months): 6-12, 12-18, 18-24, 24-30, 30-36, >36. Some patients with the event NONE had a time to event of less than six months in the training dataset. Due to this, we created an additional time window not described in the task definition, which encompasses patients with a time to event of 0-6 months.

The representation of a patient through its features differs slightly from the representation used in Section 3.2. A summary of the preprocessing and encoding steps made to each type of features (static and temporal) is given below.

## **Static Features**

- FUS, SOD1, TARDBP, C9orf72 TURIN/LISBON Merge;
- Onset bulbar, axial, generalized, and limbs converted to single categorical feature "onset", with values corresponding to the onset type;

- Limb onset type missing values imputed with value "not applicable";
- Median and mode value imputation used for missing values in other features, for numerical and categorical features respectively;
- Only kept features below 50% of missing values.

#### **Temporal Features**

- Only the ALSFRS-R total and subscores, and the forced vital capacity value were considered for the temporal features;
- The individual questionnaire questions and the start date were discarded;
- A simplistic method to deal with temporality was employed by using either the last N visits or the first N visits. In our experiments (Section 5), we experiment with  $N \in \{1, 2, 3\}$ , creating six strategies from the cartesian product between taking the first or last visits with the possible N values.

Putting it all together, the lower part of the hierarchical prognosis are classifiers, which span the multiple event types. In order to predict the (event, time to event) pair for a new patient, first, the most likely event to happen in the near future is predicted by the event prediction layer (Section 3.2).

Let us assume the predicted event was **DEATH**. Then, the patient is given to the time to event classifier that was trained with patients that have the same event exclusively, which is the **DEATH** Time to Event Classifier. The prediction from the latter is then joined with the event prediction to form a final prediction pair for the patient's prognosis.

The classifier used for each event type is empirically determined by choosing the best performing classifier from the following list: K-Nearest Neighbors, Logistic Regression, Multi-Layer Perceptron, Random Forest, AdaBoost, XGBoost and SVM's. The challenge's training dataset was split into a new training and development split with an 80/20 ratio. Hyperparameter tuning was done for each classifier with a 5-fold cross-validation grid search using the training dataset. The classifiers, with their best configuration obtained with the grid search, are tested on the development split, and the best performing classifiers for each event type are selected for the challenge.

# 4. Experimental Setup

The experiments were conducted using the datasets made available by the organization for the task 2 [22, 23].

For the event prediction (Section 3.2), the first stage of our hierarchical methodology, the experiments were run using an Apple M1 CPU with 16GB of RAM and MacOS Monterey 12.3.1.

The time to event prediction (Section 3.3) stage ran on a server with 2 Intel Xeon Silver 4216 CPU, 64GB of RAM, and Ubuntu 20.04.

The code used to learn the predictive models have been made openly available.<sup>2</sup>

All the details about the raw datasets and the used evaluation measures can be consulted in [22, 23].

<sup>&</sup>lt;sup>2</sup>https://github.com/dfmsoares/Hierarchical-Modelling-for-ALS-Prognosis

#### Table 1

Event prediction results. Random Forests were trained on 80% of the provided training data and tested on the remaining 20% through macro F1 scores.

Subtask	Data	F1 Score
А	M0	0.958
А	M6	0.967
В	M0	0.926
В	M6	0.938
С	M0	0.960
С	M6	0.954

# 5. Results

Due to the hierarchical nature of our method, its two stages / components follow different methodologies and experimental setups. As such, we will first present the results at a layer level, with their own derivative task: event prediction (Section 5.1) and time to event prediction (Section 5.2). Subsequently, we will discuss the results of the system as a whole (Section 5.3).

## 5.1. Event Prediction Results

Following the described approach for event prediction (Section 3.2), we have obtained 2 datasets per subtask, corresponding to month 0 (M0) and up to month 6 (M6) data. Model evaluation results was made through macro-averaged F1 scores, as presented in Table 1.

From the obtained results, we can extract some conclusions regarding the best strategy for this layer. First of all, the F1 scores obtained were significantly high, meaning the error rate is so small such that one can consistently predict the correct event that will happen for any given patient, leaving the remaining burden to the lower layer. In terms of the different sets of data (M0 and M6), for subtask A and B, M6 was marginally better. This is somewhat expected as the longitudinal data helps better distinguish between the possible events. For subtask C, which regards predicting only death, M0 was marginally better than M6, which could be due to the fact that in our population, clear signs are shown at the first visit which allow for a good prediction of death outcomes and some other unknown competing event.

Looking into feature importance, there was a majority of static patterns as most important features, for both M0 and M6 sets. The temporal patterns with higher feature importance mostly included the higher brackets/intervals of ALSFRS-R and its subscores. Diagnosis and onset dates, onset site, ALSFRS-R slope, weight loss over 10% and age at onset were the most commonly found factors among static patterns.

However, the performance obtained on the provided training set carries a prediction bias into the competition test set (with hidden labels). In Figure 2, we can observe the distribution of predicted events for the test set. A clear unbalance is noticed, as in the particular case of subtask A, most instances had NIV as the predicted event, with close to no examples being classified as NONE. The remaining two subtasks also had this occurring to a lesser extent.

It was observed that many instances did have non-negligible probabilities for the nonpredicted event labels. As an example, in the event model for target A using M0 data, 24.6% of test instances had over 10% probability of being labeled as DEATH, while 10.6% of instances had over 10% probability of being labeled as NONE.

These results may indicate that (1) SMOTE was not effective at dealing with imbalance, with majority events being outputted regardless; and/or (2) the patterns used by the Random Forests were most prevalent in one of the event groups, namely NIV and PEG. The first point might justify the lack of predictions for the NONE label, but not for the remaining events, which were mostly balanced in the training set. Due to the small amount of instances (and thus low absolute support), NONE was also the largest contributor in extracted patterns. However, these may also be common in the other two groups, possibly leading to few actually being informative towards predicting the minority class.



**Figure 2:** Distribution of predicted events for the competition test set in each subtask, according the different temporal contexts (M0 and M6).

## 5.2. Time to Event Prediction Results

As described in Section 3.3, the temporal context used for predictions followed a simple strategy: either selecting the first or last N visits, with N taking values from 1 to 3.

Table 2 shows the classifier with the corresponding best parameter set and selected strategy for each dataset and event type. The *S* column indicates whether the data was standardized or not. The table shows that the F1 score obtained for the best-selected classifiers is low. These results could be due to the fact that we are dealing with a non-trivial multi-class problem (6 classes) and the data unbalance is considerable. Regarding the results shown in Table 2, it can be observed that the classifiers achieve the best performance considering all the clinical visits collected, i.e. a six-month patient follow-up, except for subtask C. The best performance with M6 is expected as it gathers more information regarding each patient, from which a time-

## Table 2

Results obtained using the training set for the time to event prediction task. Strategy corresponds to the N visits selected, either the first or last N visits ( $N \in (1, 2, 3)$ ) and S indicates if the data was standardized or not.

Subtask	Strategy	Event type	S	Classifier	F1 Score
А	First 2	DEATH	Y	MLPClassifier(hidden_layer_sizes=(25, 25, 25))	0.298
А	Last 1	NONE	Υ	AdaBoostClassifier(n_estimators=25)	0.302
А	First 2	NIV	Υ	KNeighborsClassifier(metric='euclidean', n_neighbors=6)	0.317
В	Last 1	DEATH	Y	XGBClassifier(n_estimators=200)	0.336
В	First 3	NONE	Ν	LogisticRegression()	0.256
В	Last 3	PEG	Υ	MLPClassifier(hidden_layer_sizes=(25, 25, 25))	0.341
С	Last 1	DEATH	Υ	XGBClassifier(n_estimators=200)	0.290
С	First 1	NONE	Y	MLPClassifier(hidden_layer_sizes=(100, 100))	0.302

dependency can be deduced that underlines the importance of including longitudinal features in the classifiers.

For each subtask, the distribution of the time window predictions for each event type is pictured. Figure 3 shows the boxplots for Dataset A, B, and C, respectively. It is interesting to note how the prediction time changes when considering only the first visit vs all available visits up to six months, indicated as M0 and M6 respectively. In particular, the predictions for the competition test set shows that for subtask B there is a change from a PEG intervention in 18-24 months to 12-18 months later. This difference highlights the impact of the temporal features and their longitudinal time span on the perceived prognosis by the models.



Figure 3: Time to Event Predictions for the test set of each subtask (A, B and C).

## 5.3. iDPP @ CLEF 2022 Results

Tables 3 and 4 show the final results delivered by the organization, for the models predicting only the event and the pair (event, time window), respectively. We detailed the results by the macro average for recall and specificity, considering the available classes for the tasks, for the five submissions made by our team.

#### Table 3

Final iDPP challenge results with test data: Macro average recall and specificity for each submission **predicting only the event**, made by our team. M0 and M6 mean that used the available data for the first month and all the available months, respectively

Subtask	Data	Macro-Recall	Macro-Specificity
А	M0	0.498	0.749
А	M0	0.513	0.757
А	M6	0.484	0.742
А	M6	0.498	0.749
А	M6	0.500	0.750
В	M0	0.503	0.752
В	M0	0.554	0.777
В	M6	0.502	0.751
В	M6	0.494	0.748
В	M6	0.502	0.751
С	M0	0.904	0.968
С	M0	0.896	0.965
С	M6	0.887	0.962
С	M6	0.887	0.962
C	M6	0.887	0.962

Regarding the prediction of the event (regardless of the time window) we could confirm that our method achieved good results, particularly in subtask C, when considering only the first month of data (M0).

Concerning the time to event prediction, as expected by the F1 scores obtained with the trained data (see Table 2), the recall value was low. These values were somewhat expected given the high number of classes considered (16 classes, resulting from the product between events and time windows).

Figures 4, 5 and 6 show the distribution of recall and specificity values for each class, calculated from the five submissions made by our team. From the class-level measures, we can gain a few insights regarding the behavior and performance of the followed approach.

The challenge is a multi-class problem, meaning when the recall rises, specificity tends to follow the opposite direction and vice versa. Specificity is extremely high in labels where recall is low, such as any DEATH event. We expected recall to be low for these labels as we had noted beforehand that due to the extreme unbalance of the data, DEATH event predictions were scarce (Figure 2).

We can also observe that the NONE event features better recall scores, despite not being a predominant prediction.

#### Table 4

Final iDPP challenge results with test data: Macro average recall and specificity for each submission **predicting the event and the time window to event**, made by our team. M0 and M6 mean that used the available data for the first month and all the available months, respectively

Subtask	Data	Macro-Recall	Macro-Specificity
А	M0	0.203	0.904
А	M0	0.203	0.902
А	M6	0.202	0.902
А	M6	0.202	0.903
А	M6	0.202	0.902
В	M0	0.239	0.889
В	M0	0.239	0.889
В	M6	0.245	0.895
В	M6	0.244	0.896
В	M6	0.234	0.886
С	M0	0.405	0.852
С	M0	0.404	0.850
С	M6	0.412	0.857
С	M6	0.413	0.858
С	M6	0.412	0.860



**Figure 4:** Bar plot showing the mean and standard deviation scores for the recall (left) and specificity (right) of each class individually for task 2A.

An important pattern to notice is that recall for events other than NONE (e.g., NIV and PEG) tends to worsen for longer timespan predictions. The recall for time windows 12-18 is higher than for the subsequent time windows. We cannot evaluate for time windows 6-12 and >36 as their metrics were not supplied. This could be tied to two factors, which could be happening together: (i) the data unbalance is quite high, which posed a significant obstacle in this challenge; (ii) it could be that the faster the event occurs, the clearer the patterns are as the state of the patient may reflect a clear degradation of their cognitive and motor capabilities, indicating an



**Figure 5:** Bar plot showing the mean and standard deviation scores for the recall (left) and specificity (right) of each class individually for task 2B.



**Figure 6:** Bar plot showing the mean and standard deviation scores for the recall (left) and specificity (right) of each class individually for task 2C.

imminent event.

Overall, the system's performance was lower than expected; however, from the finer-grain analysis, we can gather that some time windows were easier to predict than others and posit some factors that could contribute to the global performance obtained.

# 6. Conclusions and Future Work

When diagnosed with ALS, a patient expects a life expectancy of only a few years. With no cure, treatements are focused in retarding the progression of the disease and to guarantee the quality of life for the patient. This is a challenge due to the heteregeneous nature of the disease regarding disease onset, progression rates, and survival times. Machine Learning techniques set both opportunities to improve classification tasks in ALS and challenges in development of innovative approaches and the interpretability of solutions. The objective of iDPP  $\therefore$  CLEF 2022

challenge is to design and evaluate approaches to prognostic prediction in ALS, in particular the prediction of end-stage events: initiation of non-invasive ventilation (NIV), percutaneous endoscopic gastrostomy (PEG) and death, as well as their respective time windows to occurrence.

Our work makes use of three main approaches: first, it uses an approach based on patients snapshots to group observations of the patients. In the context of ALS, this approach is useful, since patients will perform a series of clinical tests between appointments. The use of snapshots allows for a consistent description of patients conditions. Second, it uses pattern mining techniques for event prediction, which allows to use the entire patient medical history into consideration. Additionally, the patterns itself, used for classification, have a interpretability advantage, which is of particular importance in healthcare. Third, the hierarchical nature of our classification approach allows for reduced class imbalance within event data subsets and less heterogeneity across the patient population improving predictions for the less frequent (event, time window) pairs.

Within the proposed approach, the time to event prediction layer had a poor performance compared to the event layer, calling for future work on introducing methodologies which can extract more efficiently temporal data, such as deep learning or biclustering/triclustering-based classifiers [24, 17].

# Acknowledgments

This work was partially supported by Fundação para a Ciência e a Tecnologia (FCT), the Portuguese public agency for science, technology and innovation, funding to projects AIpALS (PTDC/CCI-CIF/4613/2020), LASIGE Research Unit (UIDB/00408/2020 and UIDP/00408/2020) and PhD research scholarships to DFS (2020.05100.BD), ENC (2021.07810.BD) and RTS (SFRH/BD/145377/ 2019); and by the BRAINTEASER project which has received funding from the European Union's Horizon 2020 research and innovation programme under the grant agreement No 101017598.

# References

- [1] O. Hardiman, A. Al-Chalabi, A. Chio, E. M. Corr, G. Logroscino, W. Robberecht, P. J. Shaw, Z. Simmons, L. H. van den Berg, Amyotrophic lateral sclerosis, Nature Reviews Disease Primers 3 (2017) 17071. URL: http://www.nature.com/articles/nrdp201771. doi:10.1038/ nrdp.2017.71.
- [2] J. Dorst, A. Ludolph, A. Huebers, Disease-modifying and symptomatic treatment of amyotrophic lateral sclerosis, Therapeutic Advances in Neurological Disorders (2018).
- [3] V. Grollemund, P.-F. Pradat, G. Querin, F. Delbot, G. Le Chat, J.-F. Pradat-Peyre, P. Bede, Machine Learning in Amyotrophic Lateral Sclerosis: Achievements, Pitfalls, and Future Directions, Frontiers in Neuroscience 13 (2019) 135. URL: https://www.frontiersin.org/ article/10.3389/fnins.2019.00135/full. doi:10.3389/fnins.2019.00135.
- [4] J. Ackrivo, J. Hansen-Flaschen, E. P. Wileyto, L. Schwab, Richard J.and Elman, S. M. Kawut, Development of a prognostic model of respiratory insufficiency or death in amyotrophic lateral sclerosis, European Respiratory Journal 53 (4) (2019).

- [5] R. AKueffner, N. Zach, M. Bronfeld, et al., Stratification of amyotrophic lateral sclerosis patients: a crowdsourcing approach, Sci Rep 9 690 (2019).
- [6] H. Seibold, A. Zeileis, T. Hothorn, Model-based recursive partitioning for subgroup analyses, The International Journal of Biostatistics 12 (1) (2016) 45–63.
- [7] A. V. Carreiro, P. M. Amaral, S. Pinto, P. Tomás, M. de Carvalho, S. C. Madeira, Prognostic models based on patient snapshots and time windows: Predicting disease progression to assisted ventilation in amyotrophic lateral sclerosis, Journal of biomedical informatics 58 (2015) 133–144.
- [8] A. S. Martins, M. Gromicho, S. Pinto, M. de Carvalho, S. C. Madeira, Learning prognostic models using diseaseprogression patterns: Predicting the need fornon-invasive ventilation in amyotrophic lateralsclerosis, IEEE/ACM Transactions on Computational Biology and Bioinformatics (2021).
- [9] M.-H. Soriani, C. Desnuelle, Care management in amyotrophic lateral sclerosis, Revue neurologique 173 (2017) 288–299.
- [10] S. Paganoni, C. Karam, N. Joyce, R. Bedlack, G. T. Carter, Comprehensive rehabilitative care across the spectrum of amyotrophic lateral sclerosis, NeuroRehabilitation 37 (2015) 53–68.
- [11] S. Pires, M. Gromicho, S. Pinto, M. Carvalho, S. C. Madeira, Predicting non-invasive ventilation in als patients using stratified disease progression groups, in: 2018 IEEE International Conference on Data Mining Workshops (ICDMW), IEEE, 2018, pp. 748–757.
- [12] S. Pires, M. Gromicho, S. Pinto, M. de Carvalho, S. C. Madeira, Patient stratification using clinical and patient profiles: Targeting personalized prognostic prediction in als, in: International Work-Conference on Bioinformatics and Biomedical Engineering, Springer, 2020, pp. 529–541.
- [13] E. Tavazzi, S. Daberdaku, R. Vasta, A. Calvo, A. Chiò, B. Di Camillo, Exploiting mutual information for the imputation of static and dynamic mixed-type clinical data with an adaptive k-nearest neighbours approach, BMC Medical Informatics and Decision Making 20 (2020) 1–23.
- [14] V. Grollemund, G. L. Chat, M.-S. Secchi-Buhour, F. Delbot, J.-F. Pradat-Peyre, P. Bede, P.-F. Pradat, Development and validation of a 1-year survival prognosis estimation model for amyotrophic lateral sclerosis using manifold learning algorithm umap, Scientific reports 10 (2020) 1–12.
- [15] J. Gordon, B. Lerner, Insights into amyotrophic lateral sclerosis from a machine learning perspective, Journal of Clinical Medicine 8 (2019) 1578.
- [16] J. Matos, S. Pires, H. Aidos, M. Gromicho, S. Pinto, M. de Carvalho, S. C. Madeira, Unravelling disease presentation patterns in als using biclustering for discriminative meta-features discovery, in: International Work-Conference on Bioinformatics and Biomedical Engineering, Springer, 2020, pp. 517–528.
- [17] D. Soares, R. Henriques, M. Gromicho, S. Pinto, M. de Carvalho, S. C. Madeira, Towards triclustering-based classification of three-way clinical data: A case study on predicting non-invasive ventilation in als, in: International Conference on Practical Applications of Computational Biology & Bioinformatics, Springer, 2020, pp. 112–122.
- [18] P. Fournier-Viger, Spmf: A java open-source data mining library (2008).
- [19] R. Agrawal, R. Srikant, Fast algorithms for mining association rules in large databases,

Research Report RJ 9839, IBM Almaden Research Center, San Jose, California (1994).

- [20] P. Fournier-Viger, R. Nkambou, E. Mephu Nguifo, A knowledge discovery framework for learning task models from user interactions in intelligent tutoring systems, in: Proceedings of the 7th Mexican International Conference on Artificial Intelligence (MICAI 2008). LNAI 5317, Springer, 2008, pp. 765–778.
- [21] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, J. of artificial intelligence research 16 (2002) 321–357.
- [22] A. Guazzo, I. Trescato, E. Longato, E. Hazizaj, D. Dosso, G. Faggioli, G. M. Di Nunzio, G. Silvello, M. Vettoretti, E. Tavazzi, C. Roversi, P. Fariselli, S. C. Madeira, M. de Carvalho, M. Gromicho, A. Chiò, U. Manera, A. Dagliati, G. Birolo, H. Aidos, B. Di Camillo, N. Ferro, Intelligent Disease Progression Prediction: Overview of iDPP@CLEF 2022, in: A. Barrón-Cedeño, G. Da San Martino, M. Degli Esposti, F. Sebastiani, C. Macdonald, G. Pasi, A. Hanbury, M. Potthast, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Thirteenth International Conference of the CLEF Association (CLEF 2022), Lecture Notes in Computer Science (LNCS) 13390, Springer, Heidelberg, Germany, 2022.
- [23] A. Guazzo, I. Trescato, E. Longato, E. Hazizaj, D. Dosso, G. Faggioli, G. M. Di Nunzio, G. Silvello, M. Vettoretti, E. Tavazzi, C. Roversi, P. Fariselli, S. C. Madeira, M. de Carvalho, M. Gromicho, A. Chiò, U. Manera, A. Dagliati, G. Birolo, H. Aidos, B. Di Camillo, N. Ferro, Overview of iDPP@CLEF 2022: The Intelligent Disease Progression Prediction Challenge, in: G. Faggioli, N. Ferro, A. Hanbury, M. Potthast (Eds.), CLEF 2022 Working Notes, CEUR Workshop Proceedings (CEUR-WS.org), ISSN 1613-0073, 2022.
- [24] R. Henriques, S. C. Madeira, FleBiC: Learning classifiers from high-dimensional biomedical data using discriminative biclusters with non-constant patterns, Pattern Recognition 115 (2021) 107900. URL: https://linkinghub.elsevier.com/retrieve/pii/S003132032100087X. doi:10.1016/j.patcog.2021.107900.