Predicting the Risk of & Time to Impairment for ALS patients: Report for the Lab on Intelligent Disease Progression Prediction at CLEF 2022

Aidan Mannion, Thierry Chevalier, Didier Schwab, Lorraine Goeuriot

Predicting the Risk of & Time to Impairment for ALS patients

Report for the Lab on Intelligent Disease Progression Prediction at CLEF 2022

Aidan Mannion[1,2], Thierry Chevalier[3], Didier Schwab[1] and Lorraine Goeuriot[1]

[1]*Laboratoire d'Informatique de Grenoble, Université Grenoble Alpes, CNRS, 38058 Grenoble, France*
[2]*EPOS SAS, 2-4 Boulevard Des Îles, 92130 Issy-les-Moulineaux, France*
[3]*UFR de Médecine Université Grenoble Alpes, Domaine de la Merci, 38700 La Tronche, France*

Abstract

This report details our participation at the Intelligent Disease Progression Prediction (iDPP) track at the Conference & Labs of the Evaluation Forum (CLEF) 2022. This task focuses on the progression of Amyotrophic Lateral Sclerosis (ALS), a progressive neurodegenerative disease that affects nerve cells in the brain and spinal cord. The goal of this work is to use patient demographic data & certain medical history details along with collections of records of responses to an ALS diagnostic questionnaire to calculate risk scores corresponding to the likelihood that a patient will suffer an adverse event, and to predict the time window in which that event will occur. We present an approach based on ensemble learning, in which gradient-boosted regression trees are used to separately predict risk scores and estimate survival times. By normalising & thresholding the risk scores, we generate event predictions which are combined with the time-to-event predictions to produce time-interval predictions. While some aspects of the results seem encouraging, especially given the amount of training data available, it is clear that more sophisticated and specialised solutions are required in order for techniques like these to become a reliable part of clinical decision-making.

Keywords

Clinical Data Science, Survival Analysis, Disease Progression Prediction

1. Introduction

The classification & ranking of patients according to their risk of adverse events and the estimation of when those adverse events are likely to occur are some of the most tractable & useful applications of machine learning to healthcare. Being able to efficiently and robustly predict when certain patients are likely to need urgent clinical intervention has the potential to greatly enhance the quality of care provided by healthcare professionals, from the point of view of the allocation of time & resources for consultation and for the informed construction of treatment plans.

This paper describes a proposed approach to automated prediction of the progression of Amyotrophic Lateral Sclerosis (ALS) as part of the iDPP evaluation campaign at CLEF 2022 [1, 2]. The paper is organized as follows: Section 2 introduces related works; Section 3 describes

CLEF 2022: Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy
aidan.mannion@univ-grenoble-alpes.fr (A. Mannion); thy.chevalier@free.fr (T. Chevalier); didier.schwab@imag.fr (D. Schwab); lorraine.goeuriot@imag.fr (L. Goeuriot)

our approach; Section 4 explains our experimental setup; Section 5 summarises & discusses the evaluation of the results; finally, Section 6 draws some conclusions and outlooks for future work.

The iDPP track is divided into two tasks;

- **Task 1:** Ranking Risk of Impairment for ALS - estimation of the risk of early occurrence of an adverse event and ranking subjects based on these risk scores
- **Task 2:** Predicting Time of Impairment for ALS - prediction of the time interval in which the adverse event is most likely to occur.

Three kinds of adverse event are considered; Non-Invasive Ventilation (NIV), Percutaneous Endoscopic Gastrostomy (PEG), and death. Both of the above tasks are thus divided into three subtasks, each with a different set of target variables;

1. **Task a:** NIV or death,
2. **Task b:** PEG or death,
3. **Task c:** death

Separate datasets are provided for each of these three variants. The datasets contain general patient information and some clinical history, as well as a series of observations of the progression of their disease over a six-month period, in the form of their responses to the ALSFRS-R (Amyotrophic Lateral Sclerosis Functional Rating Scale - Revised) [3], a standard method of evaluating the state of an ALS patient by how their ability to breathe, move, and perform tasks requiring muscle co-ordination is affected. Further, for each of the six tasks described, we calculate risk scores and survival times at both the beginning and the end of this six-month observation period (denoted by M0 and M6 respectively).

Data collected from clinical trials or disease progression studies tends to present many unique difficulties that inhibit the utility of standard statistical learning techniques. Most notably, clinical studies that focus on modelling the time to a particular event or patient outcome of interest almost always contain *censorship*, i.e. patients that leave the study without having experienced the event. Standard practice in machine learning would be to ignore or discard samples for which the target variable was not observed, but given the nature of clinical studies this is an impractical waste of data. The event of interest may or may not have occurred after the time of censoring, and thus it is necessary to use models that can deal with the missingness of a certain proportion of the outcome variables and use the information that is available for the censored patients.

The general class of techniques that deal with such data is known as survival analysis, which can be used to describe or compare the survival times of groups, and estimate the effect of quantitative or categorical predictor variables on survival ("survival" here being used as a general term to mean time to an event).

The survival model paradigm involves the estimation of the risk of the occurrence of an event using a *cumulative hazard function* composed of two elements;

1. an underlying risk function known as the *baseline hazard function*, which depends only on time and represents the way in which the risk of the event of interest changes at some baseline or default values for the input variables, and

2. the *effect parameters*, which are used to take the explanatory variables into account when calculating the risk score.

One of the most widely used and effective models for survival analysis is the Cox proportional hazards model [4]. *Proportional hazards* refers to the assumption that the relationship between the predictive covariates and the output of the hazard function can be effectively modelled as multiplicative. In the Cox model, the baseline hazard function $H_0(t)$ is estimated using Breslow's estimator[5], and the cumulative hazard function can thus be expressed as

$$H(t|x) = \exp(f_\theta(x))H_0(t)$$

where x is an input vector of predictive variables and $f_\theta(\cdot)$ is some parameterised multiplicative function of this input which is learned from the data.

The strategy implemented in this work is to employ Cox's proportional hazards model to the task of ranking the risk of impairment, using the gradient boosting[6] learning strategy to compute the parameters of the function $f_\theta(\cdot)$. Boosting methods in general combine the predictions of an ensemble of "weak" (i.e. very simple) decision function learners to form a more effective learning algorithm. Gradient boosting is a generalisation that allows for any differentiable loss function to be used by the ensemble. Boosting methods often entail a reduction in both bias and variance as compared to single-learner methods.

The output of the time-independent part of the survival function calculated by the gradient-boosting survival analysis method is mapped to the interval $(0, 1)$, as per the task requirements, via a sigmoid function. Any set of input subject data can then be ranked according to this function in a consistent way.

To estimate the time-to-event, we use a regression model based on Accelerated Gradient Boosting (AGB) [7]. This being a standard regression model, it does not take censoring into account. We therefore use class predictions based on the Task 1 survival model to "censor" the time-to-event predictions. More details on this process are provided in sections 3.3.1 and 3.3.2.

2. Related Work

Machine learning and survival analysis techniques have been widely experimented with in the biomedical informatics research community, in particular for the prediction of disease progression. Recent developments have of course focused on deep learning techniques - Convolutional Neural Networks (CNNs) have been particularly effective, for example in the work of Jarret et al. [8], which involved the development of a generalisable framework for the prediction of disease trajectories, or Storelli et al. [9], who used CNNs to predict Multiple Sclerosis (MS) progression from MRI data. Other advanced DL techniques have also been shown to be applicable, however, such as multimodal Recurrent Neural Networks (RNNs) for Alzheimer's progression [10], autoencoders for the prediction of head & neck squamous cell carcinoma from multi-omics data [11], or custom deep networks for the prediction of the progression of Parkinson's from telemonitoring data [12].

While deep learning is a highly promising avenue for the development of more effective computational prognostic tools that could have profound effects on the quality of care provided

to patients with these diseases, these studies all required the use of relatively large, high-dimensional, high-frequency datasets of measurements of biomedical data (multi-omics data, RNA/miRNA sequencing data, neuroimaging data, genetic biomarkers, etc.), in order to exploit the potential of deep learning. It is not clear that clinical visit data, which tends to be less detailed, lower-dimensional and more sparse than biological data, is as amenable to deep learning, particularly in the survival analysis paradigm. Indeed, certain recent studies have shown that simpler machine learning techniques such as gradient boosting are often the most effective choice for survival prediction tasks [13, 14]. Comparitive studies that use large clinical-trial datasets often show that there are many disease prediction tasks for which Support Vector Machines, Random Forest classifiers, and even Logistic Regression outperform more advanced algorithms [15, 16]. Ensemble learning techniques including random forests and gradient boosting have even shown success in the prediction of genes associated with certain diseases [17] and K-means clustering in the survival prognosis of patients with cervical cancer from genomic data [18].

In recent years the COVID-19 pandemic has attracted a wealth of research into effective prediction of disease progression from limited clinical data and with limited medical knowledge of the causal processes involved. Again, while CNNs tended to be the dominant approach for prediction of COVID severity from medical imagery [19, 20, 21], for lower-dimensional clinical data, simpler techiniques such as ensemble methods [22, 23], k-nearest neighbours[24], or logistic regression [25, 26] seem to often constitute the most effective choice.

The progression of ALS, which is the focus of this work, has long been a difficult prediction problem, due mainly to the way in which the disease exhibits a variability of onset and rate of progression, resulting in an inherent heterogeneity in the relevant clinical data. There has therefore been a wealth of investigation into the application of statistical techniques to this issue, from manual statistical analyses to determine the main clinical predictors of survival rates [27] all the way to deep learning methods based on data gathered from induced pluripotent stem cells of ALS patients [28]. Many different statistical models have been experimented with for the prediction of ALS progression from clinical trial data, including correlation analysis of genetic biomarkers [29], the Weibull statistical model [30], Kaplan-Meier survival analysis [31], the Royston-Palmer parametric survival model [32], Bayesian network classifiers [33], logistic regression [34], and random forest classifiers [35]. Given the success of Taylor et al. [35], and given that gradient-boosted decision trees often turn out to be more efficient and performant than random forest classifiers, allied with the fact that to the best of our knowledge, there is no published research testing gradient-boosting approaches on the problem of ALS progression prediction, we elected to use an approach based on gradient-boosted survival analysis.

3. Methodology

3.1. Exploratory Data Analysis

The datasets on which this work is based are of two different types; static patient information (fixed with respect to time) and visit records (varying across time). Each of the training datasets comes in three overlapping variants, one for each task; the dataset for Task A (Non-Invasive Ventilation/Death) is referred to henceforth as dataset A; likewise datasets B & C for the other

two tasks. The static-variable dataset contains 94 different variables, many of which were not used for analysis due to a very low number of discriminative samples and/or low correlation with outcome variables, as discussed in section 3.2.1. The temporal visit records contain two types of visit; ALSFRS-R questionnaire results [3], integers between 1-4 representing the answers to twelve diagnostic questions about the progression of a patient's ALS (4 being preferable), and spirometry data.

It is of particular interest in the context of this challenge to compare the results of a model that takes into account only the information available at a patient's first visit with the results obtained using all of the available visit data, which covers a period of at most six months. To give an idea of the improvement that can be expected from the addition of all visit information, we first inspect the temporal richness of the visit data. Intuitively, if there is a high proportion of patients for which there exist a relatively frequent, regular series of measurements, we could expect the temporal data to bring significant improvements to the results obtained using only the static & first-visit data.

Unfortunately, as appears to be clear from the initial analysis and as is borne out by the experimental results, the temporal visit datasets do not contain enough measurements nor enough variation in measurements over time to bring significant improvements to a statistical model. Figure 1 shows that in all three datasets, the majority of patients have ≤ 3 ALSFRS-R visits recorded in the dataset, while Figure 2 shows that even patients with a high number of visits tend not to exhibit much variation over time, making it difficult to use these temporal series to make inferences about disease progression and still more difficult for machine learning models to learn generalisable functions that can make predictions about ALS progression based on these time series. We can see that for the respiratory subscore in particular, the patients tend to give exactly the same responses at each visit, with only one of the 42 questionnaires visualised in Figure 2 having a respiratory score of 11 rather than 12.

We also found that including the spirometry data (effectively a single floating-point variable `fvc_value`) did not have any effect on classification performance in our experiments so it was excluded.

Another important aspect of the initial data analysis is to inspect the class balance in the training datasets, as highly imbalanced classification problems require adjustments to be made to the training and evaluation process. As Figure 3 shows, the outcome variable classes are reasonably balanced but there are significantly less patients that are censored (no event occurred as of the last visit on record). For this reason, sample weighting is used in the training process.

3.2. Dataset Preprocessing

3.2.1. Feature Selection

Given that our training dataset is relatively small and contains many features in the form of binary indicators, some of which only apply to a very small number of patients, it quickly became clear that the prediction task would benefit from the removal of certain features containing very little signal. The first step in the data analysis process was to attempt to identify columns in the tabular dataset that are mainly uniform and thus would not be useful for model training. Figure 4 shows that there are some binary variables in the dataset that have a reasonable (better than

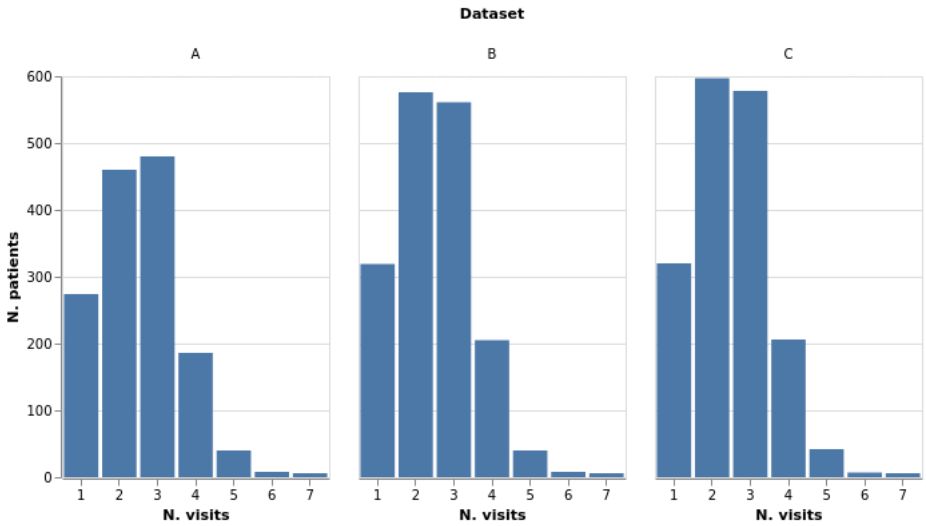

Figure 1: The count of patients in each dataset separated according to the number of ALSFRS-R questionnaire results available.

90%-10%) class balance of positive and negative cases and seem to be correlated with outcome variables, for example, in dataset A;

- while 10% of patients in total had the outcome NONE, i.e. did not die or need non-invasive ventilation during the observation period, only 4% of patients who experienced more than 10% weight loss between diagnosis and their first visit are associated with this outcome (`moreThan10PercentWeightloss` indicator),
- while 47% of the total patients required non-invasive ventilation during the observation period, 90% of the 63 who had positive values for the `major_trauma_before_onset` values, and 89% of the 135 with positive values for the `retired_at_diagnosis` indicator required NIV.

These observations seem to indicate that a patient's weight loss in the period following their diagnosis, their history of major trauma, and their retirement status (perhaps simply as a proxy for age) are strong predictive values for the outcomes of interest in this study, among other variables. Datasets B and C showed similar patterns. We found however that the very rare indicators (the long right tail of Figure 4) that refer to more specific surgical and trauma-related history, do not have enough positive examples and, empirically, do not contribute to model performance.

After initial analysis of variable prevalence and downstream experimentation on the target tasks, the input dataset to the prediction models used for evaluation uses the static patient variables shown in Table 1 (before preprocessing).

3.2.2. Feature Engineering

ALSFRS-R Responses The visit data representing the patient responses to the ALSFRS-R questionnaire are represented as individual answers to each of the 12 questions, $q_1, q_2, \ldots, q_{12}$

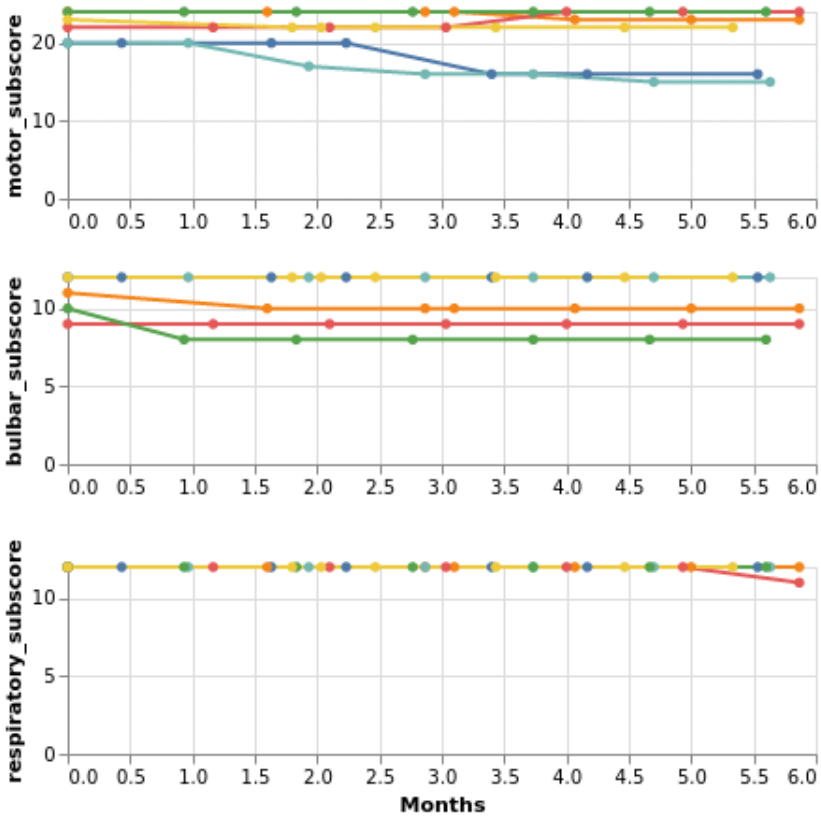

Figure 2: The change over time in the three subscores of the ALSFRS-R assessment for the six patients that completed the questionnaire seven times over a six-month observation period.

as well as the sum of the responses over three different categories of question, and the total sum. In researching the design of the ALSFRS-R questionnaire, we observed that the *motor subscore* category can be further divided into two subcategories;

- *Fine Motor Subscore:* questions 4-6,
- *Gross Motor Subscore:* questions 7-9

We therefore experimented with four different "levels" of representation of the ALSFRS-R data;

1. *Level 0:* $(d = 12)$ each question considered an individual variable $[q_i]_{i=1}^{12}$
2. *Level 1:* $(d = 4)$;
   - *Bulbar Subscore:* $S_{\text{bulb}} = q_1 + q_2 + q_3$
   - *Fine Motor Subscore:* $S_{\text{FM}} = q_4 + q_5 + q_6$
   - *Gross Motor Subscore:* $S_{\text{GM}} = q_7 + q_8 + q_9$
   - *Respiratory Subscore:* $S_{\text{resp}} = q_{10} + q_{11} + q_{12}$
3. *Level 2:* $(d = 3)$ the same as Level 1, but combining the subcategories of motor-function-related scores: $S_{\text{mot}} = S_{\text{FM}} + S_{\text{GM}}$

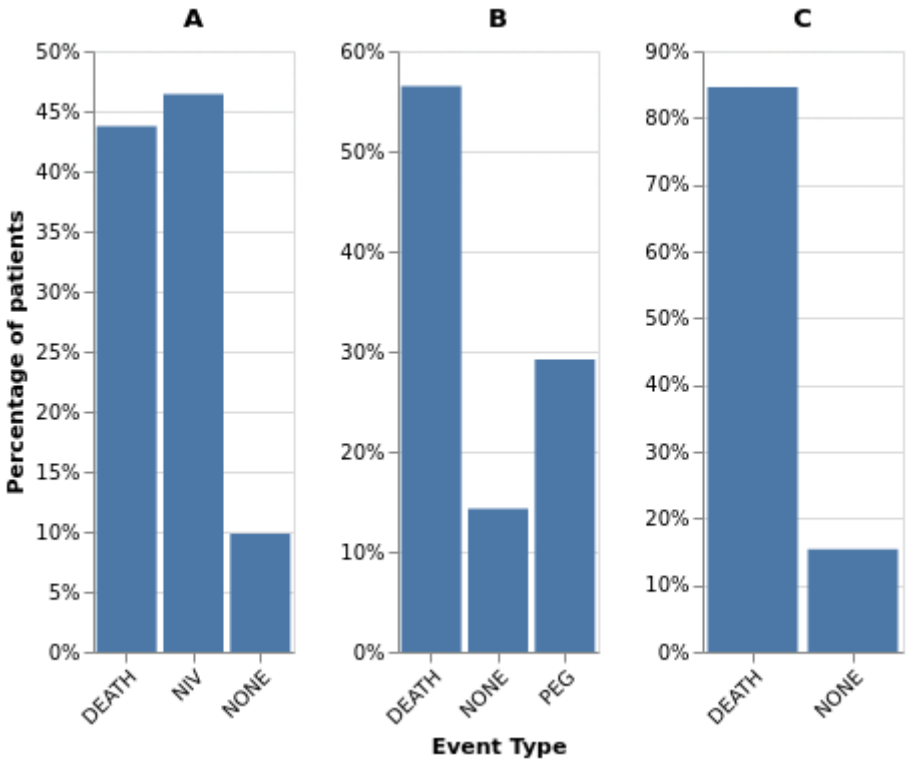

Figure 3: The percentage of patients for each target outcome in each of the training datasets.

4. *Level 3:* $(d = 1)$ using only the total ALSFRS-R score, i.e. the `alsfrs_r_tot_score` field provided.

We found in our experiments that using Level 0 as the input representation of the visit data gave the best results, likely because it gave the learning algorithm more freedom to "focus" more closely on the specific questions that were relevant to the outcome variables of interest, disregarding those that were less important - we found that adding up the responses (particularly at Level 3) serves mainly to hide the variation in individual responses.

Imputation: Weight & Height One of the first problems to be addressed with the dataset of static patient variable is that the weight & height variables (`weight`, `weight_before_onset`, and `height`), have some missing values, detailed in Table 2, with 95 patients in Dataset A, 118 patients in Dataset B and 120 patients in Dataset C missing all three of these figures.

The standard way to deal with such missingness is *multiple imputation* [36]. Imputation is the process of filling in empty fields in a tabular dataset by estimating the distribution of the non-missing values. There exist many different methods of imputation, from simple methods such as mean substitution [37], to more sophisticated statistical methods like regression imputation [38] and linear-algebraic methods such as non-negative matrix factorisation [39].

Multiple imputation is an extension of the imputation process that aims to reduce the bias

Variable Name	Type
onsetDate	float (Months)
diagnosisDate	float (Months)
sex	binary
height	float (m)
weight_before_onset	float (kg)
weight	float (kg)
moreThan10PercentWeightloss	binary
major_trauma_before_onset	binary
surgical_interventions_before_onset	binary
age_onset	float (years)
mixedMN	binary
onset_bulbar	binary
onset_axial	binary
onset_limb_type	categorical
retired_at_diagnosis	binary
ALS_familiar_history	binary
smoking	binary
turin_C9orf72_kind	categorical
hypertension	binary
diabetes	binary
dyslipidemia	binary
thyroid_disorder	binary
autoimmune_disease	binary
stroke	binary
cardiac_disease	binary
primary_neoplasm	binary
slope	float (rate of change in ALSFRS-R score since onset)

Table 1
Static variables used in the final input datasets.

Table 2
Summary of missing values from the static-variable training datasets.

Variable	Number of Missing Values		
	Dataset A (n=1454)	Dataset B (n=1715)	Dataset C (n=1756)
`height`	106	131	134
`weight_before_onset`	293	419	426
`weight`	111	141	143

and quantify the uncertainty introduced by the estimation process. This is done by averaging the outcomes across multiple imputed datasets; instead of estimating the missing values directly from the non-missing ones, multiple imputation methods estimate the underlying distribution of the dataset and creates m different imputed datasets by drawing all of the imputed values from this distribution m times. These m different imputed datasets can be used either to run downstream analysis m times, in order to quantify the uncertainty introduced by imputation through the comparison of the results, or one single imputed dataset can be estimated by taking

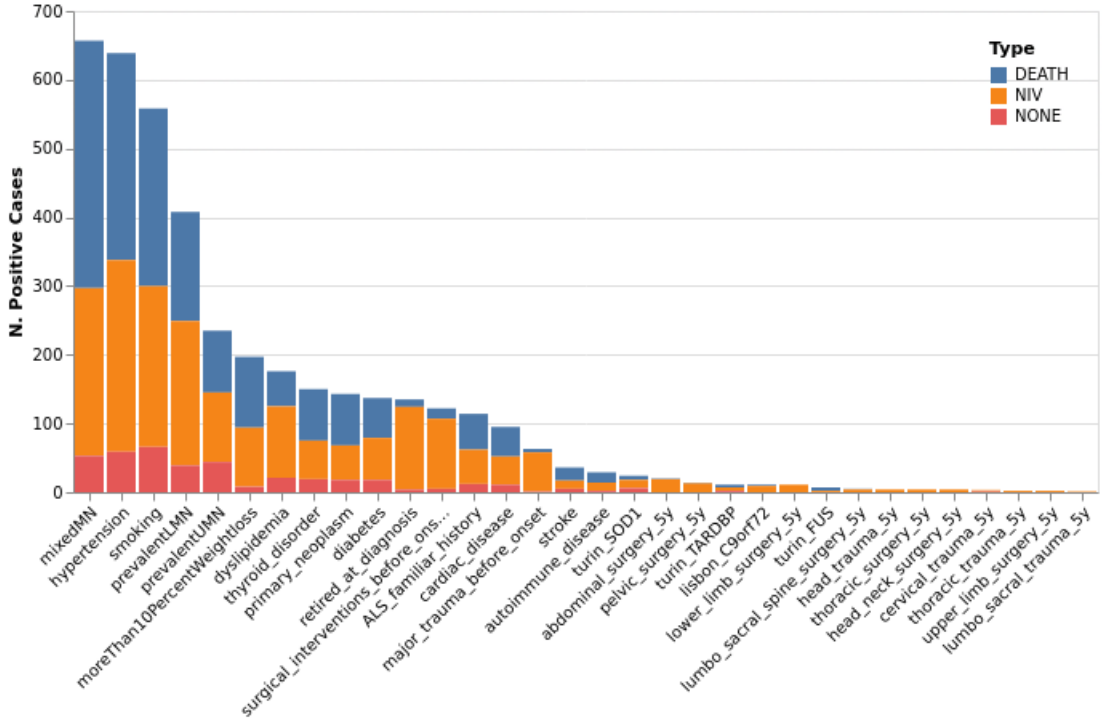

Figure 4: The number of positive examples of binary-indicator variables in dataset A (1454 patients), and the type of outcome recorded for those examples.

the average across all m datasets in order to mitigate bias that would be introduced by using single imputation.

In order to fill in the missing weight & height values, we used the MIDAS multiple imputation method [40], which is based on a denoising autoencoder architecture and has been shown to be highly effective at approximating missing values in data with complex, non-linear relationships among variables.

BMI The height and weight of a patient are not, it seems, useful health-related indicators in and of themselves, i.e. knowing a patient's height will tell us nothing about their level of risk of anything unless we know their weight, and vice versa. A commonly used way to combine information about a patient's height and weight to generate an indicator of health is the Body Mass Index (BMI), which is simply the ratio of a person's weight in kilograms to the square of their height in metres. Moreover, given that the `moreThan10PercentWeightloss` indicator seems to be an important one from the point of view of the outcome variables of interest, we decided to summarise the three variables `height`, `weight` and `weight_before_onset` with a single variable `bmi_change`, representing the change in BMI since the onset of ALS.

We also generated one-hot encodings of the categorical variables `onset_limb_type` ($d = 5$) and `turin_C9orf72_kind` ($d = 2$), resulting in an input dataset for the prediction models with 30 static variables, to be combined with the ALSFRS-R questionnaire data as described in

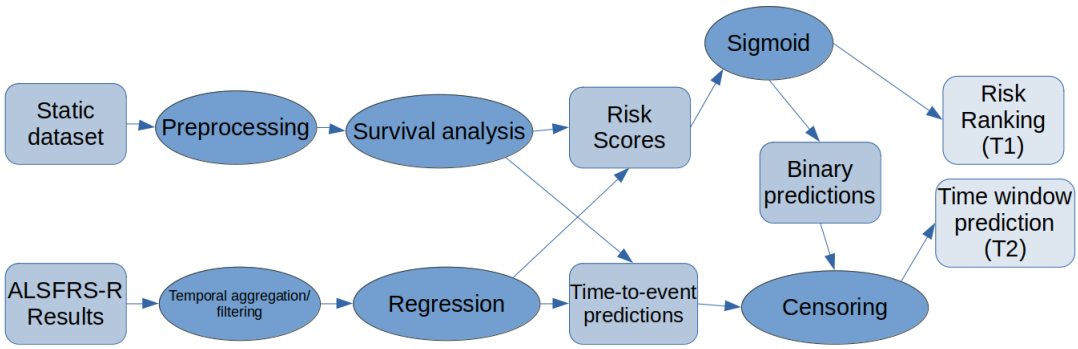

Figure 5: General overview of the prediction pipeline.

the following section. It was assumed that missing values for binary or categorical variables corresponded to "none" or negative values, and empty fields were filled in accordingly.

3.3. Combined Approach for Risk Estimation & Time-to-Event Prediction

The modelling approach taken works as follows; the questionnaire data is aggregated over time (this being for the M6 tasks; for the M0 tasks only the first visit of each patient is used) and joined to the static input variables to be used as input to two different types of gradient-boosting models;

1. binary survival-analysis models trained using outcome events as the target variable
2. regression models using the time-to-event as the target variable

The output of the first type of model can be used directly for Task 1, as described in section 3.3.1, and it's classification predictions are used in conjunction with the predictions of the regression model to form predictions for Task 2, as described in 3.3.2. An overview of the general pipeline is shown in Figure 5. For tasks A and B, the survival analysis and regression steps are run separately for the two outcomes of interest. More details on the training implementation are provided in section 4.

Temporal Aggregation Given the limitations of the temporal aspect of the visit data discussed in 3.1, it became clear that not enough data was available for more advanced sequence modelling or time-series techniques to be effective. After experimenting with several aggregation methods, we found that the best results for the M6 task were given by a simple temporally-weighted average, formulated as follows; given n scores $s_0, \ldots, s_{n-1}$, from visits taking place at times $t_0 = 0, t_1, \ldots, t_{n-1}$, we calculate the ALSFRS-R feature as

$$\sum_{i=0}^{n-1} s_i e^{t_i}$$

This allows for more recent scores to have greater proportional influence on the aggregate score.

3.3.1. Task 1: Ranking Risk of Impairment

For task 1, the target data for the training of the survival models is the event type observed for each subject; NONE or DEATH for Task 1c, along with NIV for Task 1a and PEG for Task 1b.

As the survival analysis models we use only deal with a single event type at a time, for Tasks 1a and 1b we trained two separate models; one with the DEATH examples excluded and another with the NIV/PEG examples excluded.

The issue with using a survival model "as-is" for this task is that by default, survival models assume that the event in question will indeed occur at some point for every subject. For this task, our goal is not just to rank patients according to risk, but also to associate with each patient the actual event type that is most likely to occur. Therefore, once we have chosen a risk score for each subject in a dataset, regardless of whether that score represents the risk of death, non-invasive ventilation, or percutaneous endoscopic gastrostomy, we would like to identify whether in fact the patient is more likely not to experience any adverse event at all, based on the patterns in the training dataset. This requires us to choose a classification threshold below which we label all patients as NONE. We do this based on the ROC curve for each set of predictions (again for Tasks 1a and 1b there will be two) - in each case we found it was possible to choose a threshold such that the true positive rate was greater than 0.7 and the false positive rate less than 0.5.

As the risk score outputs are theoretically unbounded, we use the sigmoid function to generate risk scores between 0 and 1, as it is a simple and effective way to map the entire real numberline into $(0, 1)$. One issue that can arise with the sigmoid is that its non-linearity does not preserve the proportions among its inputs. In particular, values outside the interval $[2 \pm \sqrt{3}]$ (points corresponding to the flexes in the sigmoid S-curve) tend to get "squished" very close together by the function, which here may affect our ability to choose an effective classification threshold. We therefore use an adapted sigmoid function with an extra parameter, denoted a, that allows the S-curve produced by the sigmoid function to be "stretched" such that the flex points occur at $\pm\frac{1}{a}$, which allows us create more space between values in a wider interval of inputs:

$$S(x, a) = \frac{1}{1 + \exp\left(-a \log\left(2 + \sqrt{3}\right) x\right)} \tag{1}$$

We experiment with different values of a to find a more optimal trade-off between true & false positive rates.

The trained risk models are used to produce a ranking of a set of subjects as follows;

1. Generate time-independent risk scores for all subjects via the trained ensembles - for Tasks 1a and 1b, we will have two models and thus two risk scores for each subject.
2. (Task 1a and 1b only) Combine the two sets of risk scores by choosing the highest one for each subject, labelling each subject with the corresponding event.
3. Map all risks to pseudo-probabilities via equation 1.
4. Choose classification thresholds based on the ROC curve for the corresponding training dataset, and label all subjects with risk scores below that value with NONE.
5. Sort the full set of subjects according to the risk scores.

3.3.2. Task 2: Predicting Time of Impairment

To predict the time window associated with the outcomes predicted by the algorithm described in the previous section, we train another gradient boosting model, this time with a one-dimensional continuous output (regression), on the same input data but with the time-to-event variables as the target instead. Having already predicted event types for each patient, we can use those predictions (DEATH, NONE, NIV, or PEG as the case may be, to "censor" the regression outputs.

The task defines six time windows into which subjects must be classified (in months); 6-12, 12-18, 18-24, 24-30, 30-36, and 36+. Thus, we use the output of the gradient-boosting regression for each subject to associate a time window with that subject, unless the predicted outcome for that patient is NONE, in which case the predicted time window is always "36+".

4. Experimental Setup

For each of the survival analysis models and regresssion models, we carried out a hyperparameter search using 5-fold cross-validation. For both model types, the variables tested in the hyperparameter search were as follows;

- `learning_rate`: scaling of the gradient descent steps; search space:

$$\left\{a \times 10^b \quad | \quad b \in [-5, \ldots, -1] \subset \mathbb{Z},\ a \in \{1, 2, 5\}\right\}$$

- `n_estimators`: number of regression trees to create; 50, 100, 150, 200
- `max_depth`: the maximum depth of each of the individual regression trees; 3, 5, or 10

The best-performing survival analysis model was selected based on the concordance index, and the regression model based on the mean absolute error, i.e. the parameterisation that gave the lowest value for the objective function on the held-out data for each cross-validation fold. Each best-performing model was retrained on the full training dataset before being saved for use on the test set.

The training pipeline was implemented in Python 3.8, using the libraries `scikit-learn` [41], `sksurv` [42], and `xgboost` [43]. Our code is made available on BitBucket[1].

For each of the 12 variants of the input data, the full hyperparameter search, metric evaluation & retraining process took around 11 minutes on 4 Intel i5-4400 3.3GHz CPUs.

5. Results

Hyperparameters The best results in the hyperparameter grid searches turned out to be given by a learning rate of 0.05 (for the survival analysis models) or 0.01 (for the regression models) and a maximum tree depth of 3, while the optimal number of estimators tended to vary from problem to problem. This uniformity is unsurprising given that each task uses many of the same training examples due to the overlap in the datasets, although overall there was very little variation in the evaluation metrics among the different hyperparameter configurations tested.

[1] https://bitbucket.org/brainteaser-health/idpp2022-lig-getalp/src/master/

Evaluation Metrics The development set metrics reported in this section ("Dev" column in the results tables) are calculated as the average over 5-fold cross-validation on the training set. It should be noted that these were calculated using different implementations of the scoring functions to those used to calculate the results on the test set. For each metric reported, we show the results on all the data on which the model was trained (to be compared to test/development results to judge the extent to which the model overfit to the training dataset), calculated using the metric calculation script provided[2], as well as our test-set results alongside the best test-set results in the evaluation campaign (both of these from the results files provided by the organisers). For more details on how the submitted results were evaluated, see the task overview [1, 2].

For Task 1, the area under the ROC curve for a censoring time of 5 years (60 months) is shown in Table 3, the Brier score for the same censoring time in Table 4 and the concordance index scores (Harrel's C-index) in Table 5. The most striking observation is that the AUROC scores are relatively poor on the development sets compared to the test set (Table 3), but this pattern is reversed in the case of the Brier score (Table 4). This is particularly unusual given that both represent scores calculated on roughly the same amount of test data (the test set is about $1/4$ the size of the training set).

Table 3
AUROC scores for Task 1 at a censoring time of 60 months.

Task	Train	Dev	Test	Best
T1a_M0	0.848	0.645	0.760	0.842
T1a_M6	0.856	0.658	0.802	0.867
T1b_M0	0.856	0.639	0.795	0.870
T1b_M6	0.855	0.641	0.811	0.877
T1c_M0	0.910	0.666	0.767	0.866
T1c_M6	0.920	0.682	0.793	0.871

Table 4
Brier scores for Task 1 at censoring time 60.

Task	Train	Dev	Test	Best
T1a_M0	0.202	0.088	0.251	0.080
T1a_M6	0.210	0.087	0.259	0.073
T1b_M0	0.199	0.102	0.217	0.106
T1b_M6	0.186	0.103	0.243	0.104
T1c_M0	0.246	0.105	0.288	0.108
T1c_M6	0.258	0.105	0.272	0.103

For the results of Task 2, we focus on the "time interval" approach to classification evaluation, i.e. we show & discuss metrics based only on the time windows predicted by the model, as opposed to the "labels" approach, where the classification target categories are the combination of the time window and the actual outcome predicted to happen in that time window, treating

[2] https://bitbucket.org/brainteaser-health/idpp2022-performance-computation/src/master/

Table 5
Concordance index scores for Task 1.

Task	Train	Dev	Test	Best
T1a_M0	0.719	0.676	0.664	0.696
T1a_M6	0.733	0.700	0.704	0.748
T1b_M0	0.736	0.705	0.694	0.725
T1b_M6	0.740	0.732	0.714	0.745
T1c_M0	0.752	0.686	0.674	0.713
T1c_M6	0.770	0.711	0.701	0.741

the evaluation as a $(c \times 5)$-label problem, where c is the number of outcomes for a given task. Given that the event we associate with each time window prediction is the same one predicted for Task 1, we found it somewhat redundant to re-evaluate these predictions in the context of Task 2, and that it is more instructive as to the performance of the time-to-event regression portion of the system to concentrate on the extent to which its predictions were within the correct time windows. Table 6 shows the average precision, or specificity, of the time window classification across all 6 possible windows, while Table 7 does the same for recall (the asterisk in these tables denotes the case where our score was the highest among the challenge participants).

The main conclusion to be drawn from the results of Task 2 is that our models, as well as the other submissions to this challenge, are much better at ensuring that the predictions that they do make are correct as opposed to finding all correct predictions in the dataset. This is deduced from the fact that specificity (precision) is much higher than recall for all models and for almost all classes. This trend suggests that survival modelling of the kind undertaken in this work can be reasonably successful at identifying risk factors for adverse events related to ALS, but can only properly process a small proportion of all the risk factors there are. We hypothesise that the models tend to miss more high-level, complex interactions between variables that constitute risks, resulting in the low recall scores.

Table 6
Macro-average specificity scores for Task 2.

Task	Train	Dev	Test	Best
T2a_M0	0.812	0.798	0.854	0.864
T2a_M6	0.782	0.663	0.850	0.876
T2b_M0	0.812	0.628	0.865	0.865*
T2b_M6	0.763	0.647	0.865	0.872
T2c_M0	0.817	0.684	0.851	0.863
T2c_M6	0.820	0.652	0.864	0.866

6. Conclusions and Future Work

This paper describes the development of a gradient boosting-based approach to the prediction of the progression of Amyotrophic Lateral Sclerosis. The goal of the work was to evaluate the effectiveness of gradient-boosting survival analysis on the ranking of ALS patients according

Table 7
Macro-average recall scores for Task 2.

Task	Train	Dev	Test	Best
T2a_M0	0.230	0.249	0.250	0.272
T2a_M6	0.220	0.338	0.216	0.341
T2b_M0	0.214	0.373	0.298	0.298*
T2b_M6	0.266	0.382	0.281	0.316
T2c_M0	0.252	0.374	0.223	0.275
T2c_M6	0.210	0.388	0.268	0.284

to their risk of impairment, and the combination of the classification outputs derived from the survival analysis with the regression outputs based on time-to-event modelling for the classification of the patients into time windows corresponding to the adverse event most likely to befall them.

The main conclusions we can draw from this work are as follows;

- Gradient-boosted survival analysis shows promise as a method for ranking patients according to risk; while the evaluation metrics are not yet satisfactory for use in a real clinical setting, performance can be expected to improve with the addition of further training data.
- The hybrid survival-classification/regression approach for time window classification seems not to be an appropriate model for time-to-event analysis in this case. We suggest other approaches to this problem below.
- Performance on the M6 task was not significantly different from that observed on the M0 task, which may suggest that ALSFRS-R measurements need to be recorded across a longer observation interval than six months, or that more detailed data is needed to create time-series in which the progression of the disease could be effectively pattern-matched.

In our estimation, advances could be made on this problem in future work in the following ways;

- Stratification of training data: it is well-known that ALS is highly variable in its rate of onset and progression, thus it is reasonable to hypothesise that it may be better to identify distince sub-populations in the training data and train separate survival models on each one, as one single analysis may not be enough to capture the complex dependencies in the dataset. This is in fact a commonly-used approach in survival analysis that can often lead to significant improvements.
- Comparison of a wider range of models: due to various constraints, this work focuses on a single learning strategy for the parameterisation of the survival and regression functions, but further work should compare this approach with random forests, support vector machines, Bayesian classifiers and neural network-based approaches. This is particularly relevant given the limitations imposed by the proportional hazards assumption, which may in fact not be ideal for a task as complex as the prediction of the progression of ALS.
- Reformulation of the time-to-event task: in this work we decided to base the training for the time-to-event prediction, which has a discrete output space, on a regression model,

which has a continuous output space. It would perhaps be more sensible and efficient, and give better results, to train the model directly on the multi-class classification task of associating each subject with a time window.

References

[1] 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.

[2] 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.

[3] J. Cedarbaum, N. Stambler, E. Malta, C. Fuller, D. Hilt, B. Thurmond, A. Nakanishi, The ALSFRS-R: a revised ALS functional rating scale that incorporates assessments of respiratory function., BDNF ALS Study Group (Phase III). J Neurol Sci. (1999). doi:10.1016/s0022-510x(99)00210-5.

[4] D. R. Cox, Regression models and life-tables, Journal of the Royal Statistical Society (1972).

[5] N. Breslow, Analysis of survival data under the proportional hazards model, International Statistical Review (1975).

[6] J. Friedman, Greedy function approximation: a gradient boosting machine, The Annals of Statistics (2001).

[7] G. Biau, B. Cadre, L. Rouvière, Accelerated gradient boosting, Machine Learning 108 (2019) 971–992.

[8] D. Jarret, J. Yoon, M. van der Schaar, Dynamic prediction in clinical survival analysis using temporal convolutional networks, IEEE Journal of Biomedical and Health Sciences (2019). doi:10.1109/JBHI.2019.2929264.

[9] L. Storelli, M. Azzimonti, M. Gueye, C. Vizzino, P. Preziosa, G. Tedeschi, N. De Stefano, P. Pantano, M. Filippi, M. Rocca, A deep learning approach to predicting disease progression in multiple sclerosis using magnetic resonance imaging, Invest Radiol. (2022). doi:10.1097/RLI.0000000000000854.

[10] G. Lee, K. Nho, B. Kang, K. Sohn, D. Kim, Predicting alzheimer's disease progression using a multi-modal deep learning approach, Sci Rep (2019). doi:10.1038/s41598-018-37769-z.

[11] Z. Zhao, Y. Li, Y. Wu, R. Chen, Deep learning-based model for predicting progression in

patients with head and neck squamous cell carcinoma, Cancer Biomark. (2020). doi:10.3233/CBM-190380.

[12] A. Hussain Shahid, M. Prasad Singh, A deep learning approach for prediction of Parkinson's disease progression, Biomedical Engineering Letters (2020). doi:10.1007/s13534-020-00156-7.

[13] A. Abuhelwa, G. Kichenadasse, R. McKinnon, A. Rowland, A. Hopkins, M. Sorich, Machine learning for prediction of survival outcomes with immune-checkpoint inhibitors in urothelial cancer, Cancers (Basel) (2021). doi:10.3390/cancers13092001.

[14] M. Konerman, L. Beste, T. Van, B. Liu, X. Zhang, J. Zhu, S. Saini, G. Su, B. Nallamothu, G. Ioannou, A. Waljee, Machine learning models to predict disease progression among veterans with hepatitis C virus, Machine Learning in Health and Biomedicine (2019). doi:10.1371/journal.pone.0208141.

[15] S. Grampurohit, C. Sagarnal, Disease prediction using machine learning algorithms, in: 2020 International Conference for Emerging Technology (INCET), 2020, pp. 1–7. doi:10.1109/INCET49848.2020.9154130.

[16] M. Ferjani, Disease prediction using machine learning (2020). doi:10.13140/RG.2.2.18279.47521.

[17] D. Le, Machine learning-base approaches for disease gene prediction, Brief Funct Genomics (2020). doi:10.1093/bfgp/elaa013.

[18] D. Ding, T. Lang, D. Zou, J. Tan, J. Chen, L. Zhou, D. Wang, R. Li, Y. Li, J. Liu, C. Ma, Q. Zhou, Machine learning-based prediction of survival prognosis in cervical cancer, BMC Bioinformatics (2021). doi:10.1186/s12859-021-04261-x.

[19] D. Haritha, N. Swaroop, M. Mounika, Prediction of COVID-19 cases using CNN with X-rays, in: 2020 5th International Conference on Computing, Communication and Security (ICCCS), 2020, pp. 1–6. doi:10.1109/ICCCS49678.2020.9276753.

[20] A. Chaddad, L. Hassan, C. Desrosiers, Deep CNN models for predicting COVID-19 in CT and X-ray images, J Med Imaging (Bellingham) (2021). doi:10.1117/1.JMI.8.S1.014502.

[21] A. A. A. Musleh, A. Y. Maghari, COVID-19 detection in X-ray images using CNN algorithm, in: 2020 International Conference on Promising Electronic Technologies (ICPET), 2020, pp. 5–9. doi:10.1109/ICPET51420.2020.00010.

[22] R. Mohammad, M. Aljabri, M. Aboulnour, S. Mirza, Classifying the mortality of people with underlying health conditions affected by COVID-19 using machine learning techniques, Applied Computational Intelligence and Soft Computing (2022). doi:10.1155/2022/3783058.

[23] S. Aljameel, I. Ullah Kahn, N. Aslam, M. Aljabri, E. Alsulmi, Machine learning-based model to predict the disease severity and outcome in COVID-19 patients, Scientific Programming (2021). doi:10.1155/2021/5587188.

[24] F. Xu, X. Chen, X. Yin, Q. Qiu, J. Xiao, L. Qiao, M. He, L. Tang, X. Li, Q. Zhang, Y. Lu, S. Xiao, R. Zhao, Y. Guo, M. Chen, D. Chen, L. Wen, B. Wang, Y. Nian, K. Liu, Prediction of disease progression of COVID-19 based upon machine learning, International Journal of General Medecine (2021). doi:10.2147/IJGM.S294872.

[25] R. Mojjada, A. Yadav, A. Prabhu, Y. Natarajan, Machine learning models for COVID-19 future forecasting, Mater Today Proc. (2020). doi:10.1016/j.matpr.2020.10.962.

[26] A. Das, S. Mishra, S. Saraswathy Gopalan, Predicting CoVID-19 community mortality risk using machine learning and development of an online prognostic tool, PeerJ. (2020). doi:10.7717/peerj.10083.

[27] T. Magnus, M. Beck, R. Giess, I. Puls, M. Naumann, K. Toyka, Disease progression in amyotrophic lateral sclerosis: predictors of survival, Muscle Nerve. (2002). doi:10.1002/mus.10090.

[28] K. Imamura, Y. Yada, Y. Izumi, M. Morita, A. Kawata, T. Arisato, A. Nagahashi, T. Enami, K. Tsukita, H. Kawakami, M. Nakagawa, R. Takahashi, H. Inoue, Prediction model of amyotrophic lateral sclerosis by deep learning with patient induced pluripotent stem cells, Annals of Neurology (2021). doi:10.1002/ana.26047.

[29] N. G. Simon, M. R. Turner, S. Vucic, A. Al-Chalabi, J. Shefner, C. Lomen-Hoerth, M. C. Kiernan, Quantifying disease progression in amyotrophic lateral sclerosis, Annals of Neurology (2014). doi:10.1002/ana.24273.

[30] R. Gomeni, M. Fava, Amyotrophic lateral sclerosis disease progression model, Pooled Resource Open-Access ALS Clinical Trials Consortium (2014). doi:10.3109/21678421.2013.838970.

[31] A.-L. Kjældgaard, K. Pilely, K. S. Olsen, A. H. Jessen, A. Ø. Lauritsen, S. W. Pedersen, K. Svenstrup, M. Karlsborg, H. Thagesen, M. Blaabjerg, Á. Theódórsdóttir, E. G. Elmo, A. T. Møller, L. Bonefeld, M. Berg, P. Garred, K. Møller, Prediction of survival in amyotrophic lateral sclerosis: a nationwide Danish cohort study, BMC Neurology (2021). doi:10.1186/s12883-021-02187-8.

[32] H. Westeneng, T. Debray, A. Visser, R. van Eijk, J. Rooney, A. Calvo, S. Martin, C. McDermott, A. Thompson, S. Pinto, X. Kobeleva, A. Rosenbohm, B. Stubendorff, H. Sommer, B. Middelkoop, A. Dekker, J. van Vugt, W. van Rheenen, A. Vajda, M. Heverin, M. Kazoka, H. Hollinger, M. Gromicho, S. Körner, T. Ringer, A. Rödiger, A. Gunkel, C. Shaw, A. Bredenoord, M. van Es, P. Corcia, P. Couratier, M. Weber, J. Grosskreutz, A. Ludolph, S. Petri, M. de Carvalho, P. Van Damme, K. Talbot, M. Turner, P. Shaw, A. Al-Chalabi, A. Chiò, O. Hardiman, K. Moons, J. Veldink, L. van den Berg, Prognosis for patients with amyotrophic lateral sclerosis: development and validation of a personalised prediction model, Lancet Neurology (2018). doi:10.1016/S1474-4422(18)30089-9.

[33] J. Gordon, B. Lerner, Insights into amyotrophic lateral sclerosis from a machine learning perspective, Journal of Clinical Medicine (2019). doi:10.3390/jcm8101578.

[34] R. Khosla, M. Rain, S. Sharma, A. Anand, Amyotrophic Lateral Sclerosis (ALS) prediction model derived from plasma and CSF biomarkers, PLoS One (2021). doi:10.1371/journal.pone.0247025.

[35] A. Taylor, C. Fournier, M. Polak, L. Wang, N. Zach, M. Keymer, J. Glass, D. Ennist, Pooled resource open-access ALS clinical trials consortium. Predicting disease progression in Amyotrophic Lateral Sclerosis, Annals of Clinical and Translational Neurology (2016). doi:10.1002/acn3.348.

[36] D. Rubin, Multiple imputation for nonresponse in surveys., 1986. doi:10.1002/9780470316696.

[37] G. Kalton, D. Kasprzyk, Imputing for missing survey responses, Proceedings of the Section on Survey Research Methods (1982).

[38] C. K. Enders, Applied Missing Data Analysis, Guilford Press New York, 2010.

[39] B. Ren, L. Pueyo, C. Chen, E. Choquet, J. H. Debes, G. Duchene, F. Menard, M. D. Perrin, Using data imputation for signal separation in high contrast imaging, The Astrophysical Journal (2020). doi:10.3847/1538-4357/ab7024.

[40] R. Lall, T. Robinson, The MIDAS touch: Accurate and scalable missing-data imputation with deep learning, Political Analysis (2021). doi:10.1017/pan.2020.49.

[41] 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, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[42] S. Pölsterl, scikit-survival: A library for time-to-event analysis built on top of scikit-learn, Journal of Machine Learning Research 21 (2020) 1–6. URL: http://jmlr.org/papers/v21/20-729.html.

[43] T. Chen, C. Guestrin, XGBoost, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2016. URL: https://doi.org/10.1145%2F2939672.2939785. doi:10.1145/2939672.2939785.