UBCS at IDPP: Predicting Patient Self-Assessment Score from Sensor Data using Machine Learning Algorithms

Notebook for the iDPP Lab on Intelligent Disease Progression Prediction at CLEF 2024

Chibuzor Chukwuemeka Okere[1,†], Edwin Thuma[1,†] and Gontlafetse Mosweunyane[1,†]

[1]*Department of Computer Science, University of Botswana, Gaborone, Botswana*

Abstract

In this paper, we explored various deep-learning techniques to develop machine learning models to perform Amyotrophic lateral sclerosis (ALS) prediction by predicting the Amyotrophic Lateral Sclerosis Functional Rating Scale-Revised (ALSFRS-R) scores for patients' second auto evaluation. To perform the task, auto-encoders and multiple imputation techniques were used to handle missing values present within the dataset. After pre-processing the data, a Random Forest algorithm was used to perform feature selection after which 4 deep neural networks predictive models were developed. The four predictive models were developed using Multi-Layer Perceptron (MLP), Feed Forward Neural Network (FFNN), Recurrent Neural Network (RNN) and Long-short Term Memory (LSTM). However, the developed models performed poorly when compared to other models in the global ranking hence, 3 more algorithms (Random Forest, Gabbing Regressor and XGBoost algorithm) were used to improve the performance of the models and the developed XGBoost algorithm outperformed other models developed in this paper as it produces minimal MAE and RMSE values.

Keywords

Amyotrophic Lateral Sclerosis (ALS), ALSFRS-R, Autoencoders, LSTM

1. Introduction

Amyotrophic lateral sclerosis (ALS) is a motor neuron disease that usually affects the nerve and spinal cord of an adult. Today, ALS is seen as a multisystem neurodegenerative disorder, a progressive neurodegenerative disease of adulthood caused by the loss of spinal, bulbar, and cortical motor neurons which can cause voluntary muscles within the human body to paralyse. [1].ALS disease typically consists of focus muscle weakness in adults from the onset. These weaknesses progress as the condition of the disease increases. This weakness usually starts in the limb muscles which is usually greater in the distal muscles when compared to the proximal muscles [2]. Currently, there is no specific cure for the disease as its condition varies from patient to patient [3]. In the early '70s and '80s, several efforts were put in by medical practitioners to develop a scale system that could be used to measure the progression of ALS in patients. These early clinical trials started with the development of a scale described by Norris in 1974 and another a decade later by Appel scale in 1984 [4]. These tools, however, were time-consuming when used hence, in the '90s a much-improved tool called the ALS functional rating scale (ALSFRS) was developed [5]. The ALS functional rating scale was designed to respond to 10 questions associated with the bulbar function, respiratory function, and upper and lower limb function of the human body [6]. For each question, a grade scale between 0 to 4 was used as 0 stands for the worst condition while 4 stands for the normal condition. A few years later, the original ALSFRS was improved and revised to allow the inclusion of the respiratory function of the human body as it is a very significant factor to be evaluated in ALS patients. The new improved scale was called ALSFRS-Revised. This new revised scale consists of 12 questions i.e. an addition of 2 more questions to the initial 10 used by ALSFRS. A healthy patient is expected to achieve a total of 48 points when this scaling system is used to evaluate the progression of ALS while the lowest score to be achieved is 0 [7]. This scaling system is relatively fast when compared to other evaluation techniques as it can be completed online or over the

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

† Authors contributed equally

moeokere17@yahoo.com (C. C. Okere); thumae@ub.ac.bw (E. Thuma); mosweuny@ub.ac.bw (G. Mosweunyane)

telephone. To improve this method a way of measuring its progression in patients, a technique called the Amyotrophic Lateral Sclerosis Functional Rating Scale (ALSFRS) was developed [3, 8, 5]. Recently, a revised version of the ALSFRS called Amyotrophic Lateral Sclerosis Functional Rating Scale-Revised (ALSFRS-R) was developed providing a more specific evaluation of the physical functionalities of the patient's well-being [9]. However, manually collecting the ALSFRS-R values is subjective and relies on self-reported data, making it prone to variability and bias which can affect its accuracy [1]. In recent years, machine learning algorithms have been utilised in the field of ALS prediction to examine the progression of ALS disease through prediction of the ALSFRS-R scores enabling the tracking of ALS progression [3]. Researchers and medical practitioners utilise machine learning algorithms in conjunction with several ALSFRS-R data, demographic and laboratory results to perform prediction of ALS progression in patients as these algorithms are capable of learning from previous patient records and making future predictions without explicitly being programmed [3, 1, 10]. The commonly used machine learning algorithms are grouped into supervised and unsupervised learning algorithms. Supervised learning algorithms are developed utilising labelled data of patient health conditions using various methods such as classification, regression and deep learning [1]. Unsupervised learning on the other hand utilises methods such as clustering and association on unlabeled patient data to examine ALS progression [11]. Commonly used machine learning algorithms for ALS prediction Include random forest, clustering algorithms, deep neural networks, and ensemble algorithms among others [1, 10, 11, 6]. These algorithms have been utilised to predict ALSFRS scores, and severity of ALS in patients and perform grouping of patients based on the progression of their ALS disease. Despite the efficiency machine-learning algorithms have brought to ALS prediction, its performance has been influenced by the high level of heterogeneity and different symptoms that exist among patients making the development of reliable, robust, and generalised machine-learning algorithms problematic. The need to develop more robust, generalised, and reliable machine learning predictive models for ALS prediction prompted the conduction of this research. Hence towards addressing this challenge, the IDPP CLEF 2024 Task 2: Predicting Patient Self-Assessment Score using Sensor Data, encourages the development of reliable machine learning predictive models for ALS prediction. In this work, we investigated the use of deep neural networks in the prediction of ALSFRS-R. In particular, we developed a deep neural network predictive model for predicting ALSFRS-R self-assessment scores of patients' second auto-evaluation by utilising sensor data collected via a mobile application. The paper is organised as follows: Section 2 presents related works done by several authors on ALS prediction; Section 3 explains the methodology we used; Section 4 explains the setup of our experiment; Section 5 presents our results and discussions on the findings made; Section 6 presents our conclusions based on these findings and possible areas for future work.

2. Related works on ALS prediction using Machine Learning

Utilising machine learning algorithms for ALS prediction has come into the limelight in recent years. Several authors have relied on the powerful ability of machine learning algorithms to develop predictive models for ALS predictions. Convolutional deep neural networks have been utilised by [1]to develop an ALS predictive model by examining the speech functionality of patients. The developed model by this author produced an AUC score of 86%. Also, another author relies on the strength of deep neural networks by comparing them with traditional algorithms for predicting ALS progression in patients. The Algorithms utilised by the authors were Convolutional Neural Networks (CNN), Feed Forward Neural network (FFNN), Recurrent Neural Networks (RNN), Feed Forward Neural network + Convolutional Neural networks (FFNN+CNN), Bidirectional and Auto-Regressive Transformer (BART), and Random Forest (RF). To examine the performance of the developed models the authors utilised Root Mean Squared Deviation (RMSD) and Pearson Correlation CCoefficient (PCC) for which FFNN+CNN outperformed other models as it produced the least RMSD and PCC scores of 0.543 and 0.415 respectively. Another author proposed a machine-learning algorithm for examining 1-year survival rate and disease progression of ALS patients by utilising a Light Gradient Boosting Machine (LGBM) and Uniform

Manifold Approximation Projection (UMAP) [11]. The developed LGBM model has an RMSE of 2.86 and an R-squared value of 0.79 while the developed U MAP algorithm has a prediction quality of 92.75%. Another study conducted by Papaiz et al. [6] introduced an Ensemble algorithm alongside 6 traditional algorithms to predict short survival rate of ALS patients using the PRO-ACT dataset. The models utilised by the authors were K-nearest Neighbour, Decision tree, random forest, support vector machine, Naive Bayes and Neural networks. Upon development and evaluation of the developed models, neural networks produced the optimal performance with an accuracy score of 88%. Despite the high accuracy of their models, the authors highlighted the need to consider using advanced algorithms called black boxes (Neural networks) as these algorithms would perform better in the development of more robust algorithms when compared to traditional algorithms. The authors also highlighted the need for effective missing value and data imbalance handling as these issues have an impact on developed models. Overall, Gordon *et. al.* [3] highlighted that utilising these machine learning techniques and algorithms can help reduce human errors and facilitating the treatment process. However, there is still a need to develop a more reliable machine-learning predictive model for ALS prediction.

3. Methodology

To perform prediction of the ALSFRS-R patient self-assessment score, we utilised various deep learning algorithms to develop predictive models. Below is a description of the dataset alongside approaches utilised during model development:

3.1. Dataset

The dataset provided for the IDPP CLEF 2024 Task 2 contains records of 52 patients [12, 13]. The dataset was grouped into 3 distinct datasets containing static data, sensor data and ALSFRS scores data of these 52 patients. Overall, the dataset for these 52 patients contains 301 self-assessed ALSFRS-R and 13946 days of sensor data. However, there are a lot of missing values within the dataset especially the sensor data due to patients failing to provide information on some days. For analysis purposes, we combined the 3 datasets using patient ID and days of diagnosis. Upon combining, the formed data set consists of 110 attributes and 7061 patient records.

3.2. Data preprocessing

One significant step we carried out was the preprocessing of data. In particular, we combined the three (3) different datasets for analysis. In addition, we performed missing values handling, feature transformation, feature selection, and feature scaling. To combine the 3 provided datasets by the IDPP CLEF 2024 Task 2 [12], we utilised the pandas library. Firstly, we combined the ALSFRS-R Data alongside the patient static data using patient ID which is a common attribute between the two datasets. This led to the formation of a single dataset containing both static data and ALSFRS-R data of the 52 examined patients. To combine the newly formed dataset with the sensor dataset, patient ID and days of diagnosis were utilised as these columns were common columns between the two datasets. The developed dataset has 7073 rows and 110 columns. However, many missing values were present in the developed dataset. Values present within the dataset that do not have corresponding key values (same patient ID and days of diagnosis) were dropped from the dataset hence the final developed dataset contained 299 rows and 110 columns. To handle missing values still present within the well-matched dataset, we utilised two techniques using autoencoders and multiple imputation techniques as provided in the mice forest library [14]. These two methods are more powerful as compared to traditional mean and median way of handling missing data. This is because the said traditional methods are prone to outliers and don't check for relationship from other variables present in the dataset. Auto encoders and Multiple Imputation check for relationship between variables thereby reducing outliers and noise. The main reason for combining both methods is because auto encoders can be biased while multiple imputation has computational complexity. So the idea is to use auto encoders to minimize missing

values and then multiple imputation handles the bias therefore giving a robust, scalable and flexible technique of handling missing values. So basically, these two methods complement each other in handling missing values. For feature transformation, categorical data present within the dataset were converted to categories using a label encoder. To select features, a Random forest algorithm was used to sort features based on their importance [1]. The selected features that provide optimum performance upon the development of models were 10 in total depicted in Figure 1. Upon selecting features, we performed scaling of the selected features using MinMaxscaler and StandardScaler libraries provided by the sklearn library for normalisation and standardisation of data.

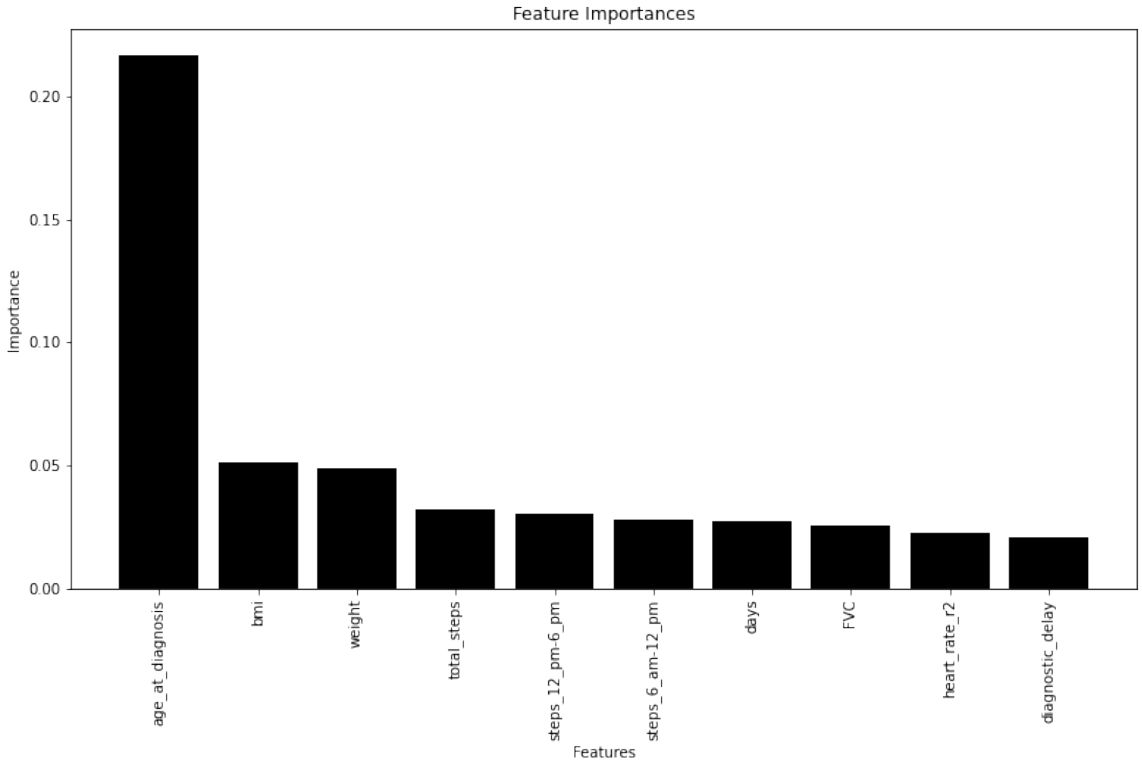

Figure 1: Selected Features using Random Forest Algorithms

3.3. Algorithm Utilised

For our investigation, 4 deep neural networks were used for model development. These neural networks were multi-layer perception (MLP), feed-forward neural networks (FFNN), recurrent neural network (RNN) and Long-short Term Memory Neural networks (LSTM) [1]. The structure of a neural network algorithm is depicted in Figure 2. Deep neural works consist of input layers, hidden layers and output layers. For a prediction to be done, input data are fed to the input layer which passes it to the hidden layer through the weights of the neurons [15]. This hidden layer does the majority of the work within the deep neural network architecture and then finally passes its predictions to the output layer [1]. The MLP and LSTM are feed-forward algorithms while the RNN and LSTM are recurrent neural network algorithms that work by propagating backwards to update weights of neural networks to reduce class weight to reduce loss or error of a developed model [16]. All developed deep neural networks consist of 1 input layer containing neurons based on the number of selected features, 5 hidden layers (4 dense layers and 1 dropout layer), and an output layer expected to produce 12 outputs. We utilised the relu activation function for both input and output layers and the linear activation function of the output layer as the task at hand is a regression problem.

In this work, we used mean absolute error (MAE) and Root Mean Squared Error (RMSE) to evaluate the performance of each developed model [18]. Evaluating the performance of a machine learning

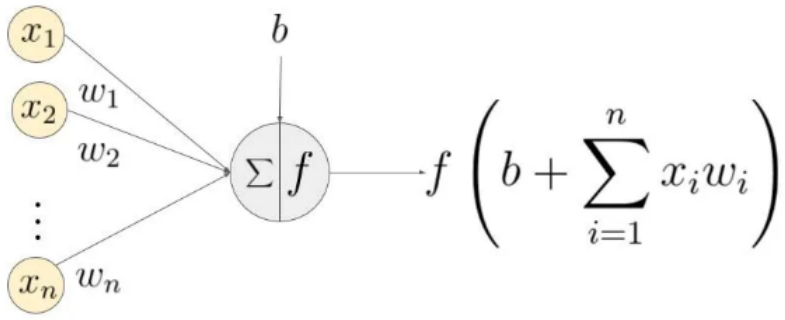

Figure 2: Structure of a Neural Network Algorithm [17].

model is crucial in machine learning as it tells how well a model is performing predictions [18]. MAE are used to calculate the error of an algorithm as the evaluation technique can handle outliers resent within a dataset RMSE however is sensitive to outliers and penalises errors more than RMSE due to the squaring done when generating the mean squared error of an algorithm Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) [19]. MAE and RMSE are calculated using equations 1 and 2 shown below:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|e_i|$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=0}^{n}|e_i|^2}$$

Where n = Total number of samples
i = each sample in the dataset
e = error of each instance

4. Experimental Setup

4.1. Repository Organization

The repository is organized into 4 sections containing the code, scores, submissions, resources and report. The code folder contains all Jupyter files developed during model development, the submission folder contains all the runs we developed, the scores folder contains all the scores of developed models and the resource folder holds other developed files during model development such as exported best-performing models.

4.2. Tools used

Tools utilised for the experiment include:

1. Toshiba Satellite c850 Laptop (core i5 8gig Ram 2.5 GHZ)
2. Jupyter Notebook
3. Libraries used (pandas, sklearn, Tensorflow, mice library, matplotlib, numpy and pickle)

4.3. Training and Validation dataset

We split our dataset into training and validation splits in the ratio of 90 to 10%

4.4. Description of run

For this experiment, 6 runs were developed based on feature selection. The runs are depicted in Table 1 below. Among the runs, run 1 which contains 10 features (age_at_diagnosis, bmi, weight, total_steps, steps_12_pm-6_pm, step_6_am-12_pm, days, FVC, heart_rate_r2 and diagnostic_delay) outperformed other runs hence it was picked for further improvement of the developed model.

Table 1
Runs Description

Runs	Run name	Description
Run 1	Ubcs_T2_features10	In this run, the top 10 features from the dataset were selected using the random forest algorithm to develop the deep learning models for ALSFRS-R prediction
Run 2	Ubcs_T2_features20	In this run, the top 20 features from the dataset were selected using the used random forest algorithm to develop the deep learning models for ALSFRS-R prediction
Run 3	Ubcs_T2_features25	In this run, the top 25 features from the dataset were selected using the used random forest algorithm to develop the deep learning models for ALSFRS-R prediction
Run 4	Ubcs_T2_features50	In this run, the top 50 features from the dataset were selected using the used random forest algorithm to develop the deep learning models for ALSFRS-R prediction
Run 5	Ubcs_T2_features100	In this run, the top 100 features from the dataset were selected using the used random forest algorithm to develop the deep learning models for ALSFRS-R prediction
Run 6	Ubcs_T2_featuresall	In this run, all features in the dataset were selected using the used random forest algorithm to develop the deep learning models for ALSFRS-R prediction

5. Results and Discussion

Table 2
IDPP CLEF 2024 submission rankings

Rank	run	MAE	RMSE
1	fcool	0.2878	0.5774
2	unipd	0.2878	0.5774
3	Compbiomedunito_randomforest_MW	0.3000	0.6013
4	bitua	0.3257	0.6086
5	bitua	0.3712	0.6542
. . .	. . .	. . .	. . .
25	Ubcs_T2_features25	0.9624	1.1367
26	Ubcs_T2_features20	1.0238	1.1808
27	Ubcs_T2_features100	0.8214	1.1965
28	Ubcs_T2_featuresall	0.8918	1.2516
29	Ubcs_T2_features10	0.9404	1.3331
31	Ubcs_T2_features50	1.1071	1.5238

The results depicted in Table 2 show the global ranking of the IDPP CLEFF 2024 competition for Task 2 which focuses on developing a machine learning predictive model for ALSFRS-R value for the patient's

second assessment. Our submissions (which occupy positions 25 to 29 and 31 denoted using Ubcs) were ranked low among other submissions as they produced higher RMSE and MAE when compared to models developed by other teams. These low performances by our developed models can be attributed to the fact that deep neural networks are not being suitable for the task at hand as other algorithms. In particular, ensemble algorithms used by other teams outperformed the deep neural networks. To improve our developed model, we conducted further analysis by utilising 3 ensemble algorithms which were Random Forest, Bagging Regressor, and XGBoost algorithm to improve our work. Ensemble algorithms are a group of machine learning algorithms formed through the combination of different learning models to produce a single model with better performance. Common types of ensemble learning algorithms include Bagging algorithms, Extreme Gradient Boosting algorithms, Random forest, Stacking algorithms [20]. The developed ensemble models gave improved performance similar to that of highly ranked teams. Among the developed ensemble predictive models, It can be seen in Figure 3 and Table 3 that the XgBoost algorithm gave a higher performance on run 1 as it produced an MAE value of 0.413889 and an RMSE value of 0.747217 indicating that ensemble algorithms have the potential to outperform deep neural networks on the given task.

ENSEMBLE MODELS

```
from sklearn.ensemble import RandomForestRegressor,BaggingRegressor
from xgboost import XGBRegressor
from sklearn.tree import DecisionTreeRegressor
rf_model=RandomForestRegressor()
rf_model.fit(x_train,y_train)
rf_pred=rf_model.predict(x_test)
rf_pred=np.round((np.clip(rf_pred,0,4))).astype(int)
print('MAE for RandomForest:',mean_absolute_error(rf_pred,y_test))
print('RMSE for RandomForest:',np.sqrt(mean_squared_error(rf_pred,y_test)))
```

```
MAE for RandomForest: 0.44999999999999996
RMSE for RandomForest: 0.7601169500660919
```

```
bg_model=BaggingRegressor(estimator=DecisionTreeRegressor())
bg_model.fit(x_train,y_train)
bg_pred=bg_model.predict(x_test)
bg_pred=np.round((np.clip(bg_pred,0,4))).astype(int)
print('MAE for Bagging Model:',mean_absolute_error(bg_pred,y_test))
print('RMSE for Bagging Model:',np.sqrt(mean_squared_error(bg_pred,y_test)))
```

```
MAE for Bagging Model: 0.49444444444444446
RMSE for Bagging Model: 0.8232726023485646
```

```
xg_model=XGBRegressor()
xg_model.fit(x_train,y_train)
xg_pred=xg_model.predict(x_test)
xg_pred=np.round((np.clip(xg_pred,0,4))).astype(int)
print('MAE for xgboost:',mean_absolute_error(xg_pred,y_test))
print('RMSE for xgboost:',np.sqrt(mean_squared_error(xg_pred,y_test)))
```

```
MAE for xgboost: 0.4138888888888889
RMSE for xgboost: 0.7472170590486632
```

Figure 3: Improvement using Ensemble Algorithms

Table 3
Perfomance Using Ensemble Algorithms

Model	MAE	RMSE
RandomForest	0.450000	0.760117
Bagging Algorithm	0.49444	0.823273
XgBoost	0.413889	0.747217

6. Conclusion and Future work

In this paper, deep neural networks were utilised to develop models for predicting ALSFRS-R self-assessment scores for patients. Despite how powerful deep neural networks are, the results achieved in this work showed that other algorithms such as ensemble algorithms have the potential to outperform deep neural networks. This was seen as we attempted to improve the model and utilised ensemble algorithms which produced better performance than the initially developed predictive models using deep neural networks. The dataset size used during model training also plays a major role in the low performance of the developed model. In future work, sufficient data can be utilised for training to improve the performance of the developed models. Also, hyper-parameter fine-tuning techniques such as GridsearchCV alongside cross-validation techniques can be used on developed models to improve performance.

References

[1] C. Pancotti, G. Birolo, C. Rollo, T. Sanavia, B. Di Camillo, U. Manera, A. Chiò, P. Fariselli, Deep learning methods to predict amyotrophic lateral sclerosis disease progression, Scientific reports 12 (2022) 13738.

[2] P. Masrori, P. Van Damme, Amyotrophic lateral sclerosis: A clinical review, European journal of neurology 27 (2020) 1918–1929.

[3] J. Gordon, B. Lerner, Insights into amyotrophic lateral sclerosis from a machine learning perspective, Journal of Clinical Medicine 8 (2019) 1578.

[4] S. B. Rutkove, Clinical measures of disease progression in amyotrophic lateral sclerosis, Neurotherapeutics 12 (2015) 384–393.

[5] F. G. Vieira, S. Venugopalan, A. S. Premasiri, M. McNally, A. Jansen, K. McCloskey, M. P. Brenner, S. Perrin, A machine-learning based objective measure for als disease severity, NPJ digital medicine 5 (2022) 45.

[6] F. Papaiz, M. E. T. Dourado Jr, R. A. de Medeiros Valentim, R. Pinto, A. H. F. de Morais, J. P. Arrais, Ensemble-imbalance-based classification for amyotrophic lateral sclerosis prognostic prediction: Identifying short-survival patients at diagnosis, BMC Medical Informatics and Decision Making 24 (2024) 80.

[7] B. Ilse, T. Prell, M. Walther, V. Hartung, S. Penzlin, F. Tietz, O.-W. Witte, B. Strauss, J. Grosskreutz, Relationships between disease severity, social support and health-related quality of life in patients with amyotrophic lateral sclerosis, Social Indicators Research 120 (2015) 871–882.

[8] J. D. Berry, S. Paganoni, K. Carlson, K. Burke, H. Weber, P. Staples, J. Salinas, J. Chan, J. R. Green, K. Connaghan, et al., Design and results of a smartphone-based digital phenotyping study to quantify als progression, Annals of clinical and translational neurology 6 (2019) 873–881.

[9] N. Zhou, P. Manser, Does including machine learning predictions in als clinical trial analysis improve statistical power?, Annals of Clinical and Translational Neurology 7 (2020) 1756–1765.

[10] H. Turabieh, A. S. Afshar, J. Statland, X. Song, T. P. R. O.-A. ALS, C. T. Consortium, Towards a machine learning empowered prognostic model for predicting disease progression for amyotrophic lateral sclerosis, in: AMIA Annual Symposium Proceedings, volume 2023, American Medical Informatics Association, 2023, p. 718.

[11] T. Anani, J.-F. Pradat-Peyre, F. Delbot, P.-F. Pradat, Machinelearning and feature selection methods to predict 1-year survival and disease progression for amyotrophic lateral sclerosis (2024).
[12] G. Birolo, P. Bosoni, G. Faggioli, H. Aidos, R. Bergamaschi, P. Cavalla, A. Chiò, A. Dagliati, M. de Carvalho, G. M. Di Nunzio, P. Fariselli, J. Manuel, G. Dominguez, M. Gromicho, A. Guazzo, E. Longato, S. C. Madeira, U. Manera, S. Marchesin, L. Menotti, G. Silvello, E. Tavazzi, E. Tavazzi, I. Trescato, M. Vettoretti, B. Di Camillo, N. Ferro, Overview of idpp@clef 2024: The intelligent disease progression prediction challenge, in: Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), Grenoble, France, September 9th to 12th, 2024, CEUR Workshop Proceedings, CEUR-WS.org, 2024.
[13] G. Birolo, P. Bosoni, G. Faggioli, H. Aidos, R. Bergamaschi, P. Cavalla, A. Chiò, A. Dagliati, M. de Carvalho, G. M. Di Nunzio, P. Fariselli, J. Manuel, G. Dominguez, M. Gromicho, A. Guazzo, E. Longato, S. C. Madeira, U. Manera, S. Marchesin, L. Menotti, G. Silvello, E. Tavazzi, E. Tavazzi, I. Trescato, M. Vettoretti, B. Di Camillo, N. Ferro, Intelligent Disease Progression Prediction: Overview of iDPP@CLEF 2024, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction - 15th International Conference of the CLEF Association, CLEF 2024, Grenoble, France, September 9-12, 2024, Proceedings, 2024.
[14] X. Dai, Evaluating the Performance of Different Multiple Imputation Methods When Imputing Missingness in Time-Series-Cross-Sectional Data, Ph.D. thesis, University of Washington, 2019.
[15] P. Sharma, S. K. Dinkar, D. Gupta, A novel hybrid deep learning method with cuckoo search algorithm for classification of arrhythmia disease using ecg signals, Neural computing and Applications 33 (2021) 13123–13143.
[16] P. Murugan, Feed forward and backward run in deep convolution neural network, arXiv preprint arXiv:1711.03278 (2017).
[17] T. Babs, The mathematics of neural networks, Medium, July (2018).
[18] T. Chai, R. Draxler, A. Stein, Source term estimation using air concentration measurements and a lagrangian dispersion model–experiments with pseudo and real cesium-137 observations from the fukushima nuclear accident, Atmospheric Environment 106 (2015) 241–251.
[19] M. Padhma, A comprehensive introduction to evaluating regression models, Data Science Blogathon updated On October 31st (2023).
[20] A. Soni, K. P. Shankar, Bank loan default prediction using ensemble machine learning algorithm, in: 2022 Second International Conference on Interdisciplinary Cyber Physical Systems (ICPS), IEEE, 2022, pp. 170–175.