Auto-Sklong: A New AutoML System for Longitudinal Classification

Simon Provost
University of Kent
Canterbury, United Kingdom
simon.gilbert.provost@gmail.com

Alex A. Freitas
University of Kent
Canterbury, United Kingdom
a.a.freitas@kent.ac.uk

***Abstract*—Automated Machine Learning (AutoML) addresses the challenge of selecting the best machine learning algorithm and its hyperparameter settings for a given dataset. However, existing AutoML systems typically focus on standard classification tasks and cannot directly exploit temporal information e.g. in longitudinal datasets, which contain multiple measurements of the same features over time — a common scenario in biomedical applications. We introduce Auto-Sklong, the first AutoML system that includes longitudinal classification algorithms in its search space. Experiments with 20 age-related disease datasets from the English Longitudinal Study of Ageing demonstrate that Auto-Sklong significantly outperforms a state-of-the-art AutoML system (Auto-Sklearn) and two baseline random forest methods in terms of predictive accuracy.**

***Index Terms*—AutoML, Longitudinal Data, Classification, Supervised Machine Learning, Age-Related Diseases**

I. Introduction

Longitudinal datasets contain information about the same cohort of individuals over time, with features repeatedly measured across different time points (also called "waves") [1], [2]. Analysing such data on age-related diseases is crucial due to the increasing proportion of elderly populations worldwide, which strains healthcare and socioeconomic systems.

Standard supervised machine learning (ML) methods are not tailored for longitudinal data on patients' health trajectories [3], highlighting the need for new ML methods designed for such data. We focus on the longitudinal classification task, aiming to predict a class variable (e.g., presence or absence of an age-related disease) based on longitudinal features.

There are two broad approaches to applying classification algorithms to longitudinal data: (1) *data transformation*, where longitudinal data are converted into standard non-longitudinal format, allowing the use of standard classification algorithms; and (2) *algorithm adaptation*, where classification algorithms are adapted to directly handle longitudinal data, leveraging temporal information to improve accuracy [2], [4].

Hence, there are currently several options for applying classification algorithms to longitudinal data, and an important open question is: What is the best classification algorithm and hyperparameter settings for a given input dataset? This question is central to Automated ML (AutoML), a sub-area of ML with numerous systems proposed in the literature [5], [6].

AutoML systems search for the best classification pipeline (including algorithm and preprocessing methods) and their best hyperparameter settings by iteratively evaluating candidate pipelines' predictive accuracies. This search often outperforms manual algorithm selection. However, existing AutoML systems generally address standard classification tasks [5] and cannot directly exploit temporal information in longitudinal data. Applying them to longitudinal data requires data transformation, which may be suboptimal given the availability of algorithms that directly handle longitudinal data [4].

To address this gap, we propose Auto-Scikit-Longitudinal (Auto-Sklong), which, to the best of our knowledge, is the first AutoML system with a search space that includes longitudinal classification algorithms. Developing an AutoML system for longitudinal classification is more challenging than for standard classification because it should encompass both data transformation and algorithm adaptation approaches, to be more flexible and provide more options for the search method.

Auto-Sklong follows this flexible approach by including in its search space classification algorithms and feature selection methods based on both data transformation and algorithm adaptation. We evaluated Auto-Sklong on 20 real-world longitudinal datasets from the English Longitudinal Study of Ageing (ELSA) [7], involving combinations of two types of biomedical features (Nurse and Core data) and 10 age-related diseases as binary class variables. We compared Auto-Sklong with three approaches: (a) Auto-Sklearn [8], a state-of-the-art AutoML system for standard classification; (b) standard random forest (RF) [9]; (c) a longitudinal version of RF [10].

The results have shown that the Auto-Sklong achieves in general significantly better predictive accuracy than Auto-Sklearn and the two versions of RF used as baselines.

The remainder of the paper is organised as follows. Section II describes the background and related work. Section III introduces our proposed AutoML system. Section IV describes the experimental setup. Section V reports the computational results and their discussion. Finally, Section VI summarises our findings and suggests future work.

II. Background

A. Longitudinal Classification

Longitudinal classification is a variant of supervised learning where features are measured at multiple time points (waves) [2], e.g., cholesterol levels measured over several waves. It is particularly relevant in biomedical applications, as patient data are often collected over long time periods.

The goal is to learn a model that predicts the class label (Y) for an instance while accounting for the evolution of feature values over time, i.e., to learn a classifier function of the form:

$$\begin{aligned} Y \leftarrow f(&X_{1,1}, X_{1,2}, \ldots, X_{1,T}, \\ &\vdots \quad \vdots \quad \vdots \\ &X_{J,1}, X_{J,2}, \ldots, X_{J,T}) \end{aligned}$$

where $X_{i,j}$ is the value of the i-th feature at the j-th wave, $i = 1, \ldots, J$, $j = 1, \ldots, T$, with J features and T waves. The classifier function $f(\cdot)$ must account for temporal dependencies among feature values. Note that the features are longitudinal, but the class variable is not; the goal is to predict the class label at a single wave (usually the most recent wave).

There are two main approaches for handling longitudinal data [2]. The first is *data transformation*, where longitudinal data are transformed into standard, "flattened" non-longitudinal data, allowing the use of standard classification algorithms. However, this may lead to loss of information about temporal changes. The second approach is *algorithm adaptation*, which involves using classification algorithms designed to directly handle temporal variations in features. Both approaches are used in the proposed AutoML system.

B. Automated Machine Learning (AutoML)

AutoML addresses the Combined Algorithm Selection and Hyperparameter (CASH) optimisation problem [11], which can be formulated as a bi-level optimisation problem [12]. The upper level selects the best ML algorithm or pipeline (including data preprocessing methods), and the lower level optimises hyperparameters for those algorithms/methods, aiming to minimise a predefined loss function on a validation set.

Consider an input dataset D, divided into a training set D_{train} and a validation set D_{val}. The objective of an AutoML system is to minimise a predefined loss function, $L(.)$. For classification tasks, this might be e.g. the Area Under the Receiver Operating Characteristic curve (AUROC) [13].

The *search space* of an AutoML system usually consists of several classification algorithms and data pre-processing methods (forming a classification pipeline), and an AutoML system aims at solving the following optimisation equation:

$$\begin{aligned} A^*_{\lambda^*} &= \underset{A^{(i)} \in \mathcal{A}, \lambda \in \Lambda^{(i)}}{\arg\min} \; \mathcal{L}_{val}(A^{(i)}_{\lambda,w^*}, D_{val}), \\ \text{s.t.} \quad w^* &= \underset{w}{\arg\min} \, \mathcal{L}_{train}(A^{(i)}_{\lambda,w}, D_{train}), \end{aligned} \tag{1}$$

where A^* is the optimal algorithm with its optimal hyperparameter settings λ^*, and w^* are the parameters of the model learnt by A^* with λ^*. The system searches for the algorithm and hyperparameter settings that minimise $\mathcal{L}_{val}$, the loss on D_{val}, with the algorithm trained to minimise $\mathcal{L}_{train}$, the loss on D_{train}. Each i-th algorithm $A^{(i)}$ has its own hyperparameter space $\Lambda^{(i)}$, and the i-th candidate solution is denoted $A^{(i)}_{\lambda,w}$.

A *search method* navigates the search space. Bayesian optimisation (BO) is a popular iterative method for optimising computationally intensive black-box functions [14], [15]. BO balances exploration and exploitation using a surrogate model to approximate the relationship between an ML pipeline's configuration and its predictive accuracy. It employs an acquisition function, such as Expected Improvement (EI) [16], to select the next configuration, iterating until a stopping criterion is met, and it returns the best solution found, $A^*_{\lambda^*,w^*}$.

C. Related Work on AutoML for Longitudinal Data

Most existing AutoML systems focus on standard (non-longitudinal) classification tasks, requiring longitudinal data to be transformed into standard tabular format before application. This transformation can be suboptimal due to the loss of temporal information.

An exception is the work in [17], which proposes a BO method for data where both the features and the class variables are longitudinal. This method learns a classifier for each time point and uses a new acquisition function that considers classifiers' accuracies across all time points, increasing the likelihood of selecting classifiers that perform well overall.

The BO in [17] differs from the AutoML system proposed in this work (Auto-Sklong, see Section III) in four major ways: (1) it learns separate "local" classifiers for each time point by flattening the data, whereas Auto-Sklong learns a "global" longitudinal classifier from all time points; (2) its search space lacks longitudinal classification algorithms, while Auto-Sklong's search space contains several such algorithms; (3) it requires both features and class variables to be longitudinal; whereas Auto-Sklong handles data where only features are longitudinal and the class variable is available at a single time point (as in our experiments); (4) it addresses only hyperparameter optimisation, while Auto-Sklong addresses the full CASH problem, also including algorithm selection.

There are AutoML systems for multivariate time-series data [3], such as AutoGluon-TimeSeries [18]. However, there are some differences between time-series and longitudinal data. In particular, time-series data typically contain only numerical variables and many time points; whereas longitudinal data typically include both numerical and categorical variables (particularly in biomedical applications) and have few time points (e.g., 4–8 in our datasets). Thus, some techniques often used in time-series data, such as sliding windows, are generally not suitable for longitudinal data. Hence, time-series AutoML systems are not discussed further in this work.

III. Auto-Sklong: A Novel AutoML System Tailored For Longitudinal Classification

This section introduces Auto-Sklong, an AutoML system specifically designed for longitudinal classification. It is — to the best of our knowledge — the first AutoML system whose search space includes both longitudinal and standard (non-longitudinal) classification algorithms, the latter applied to longitudinal data via data transformation. Importantly, Auto-Sklong determines whether traditional or longitudinal methods are better suited for the input dataset by evaluating their performance during an iterative search process.

A. Constructing Classification Pipelines based on Auto-Sklong's Sequential Search Space

Auto-Sklong iteratively creates and evaluates candidate classification pipelines based on a search space of ML algorithms and feature selection methods, each with candidate hyperparameter settings. Each candidate solution is built in three sequential steps by selecting: (1) a data preparation approach; (2) a feature selection method (or none); and (3) a classification algorithm. Each step's choices depend on previous steps. We describe the options in each step below. An overview of Auto-Sklong's search space is shown in Figure 1.

Step 1 – Data Preparation: Auto-Sklong randomly selects either the data transformation or algorithm adaptation approach, each with a 50% chance, which determines the available methods in Steps 2 and 3. The algorithm adaptation approach, called MerWavTime(+) ("Merge Waves and keep features' Time indices") in [2], preserves temporal information by maintaining the time indices of all features [19]–[21]; allowing the use of longitudinal methods from our Sklong library to learn temporal patterns.

The data transformation approach converts longitudinal data into standard tabular form, limiting Steps 2 and 3 to use non-longitudinal methods from Scikit-Learn [8]. Auto-Sklong randomly selects one of 10 data-flattening methods, categorised into three groups, viz.:

(a) Using an aggregation function (mean or median) to replace all values of a longitudinal feature across waves with a single value [22], [23], producing a single-wave dataset.

(b) Merging all features from all waves into a single set, disregarding time indices; different values of a feature across time points are treated as distinct features [24], [25]. This approach is called MerWavTime(–) ("Merge Waves and discard features' Time indices") in [2].

(c) Treating each wave as a separate dataset, learning a classifier for each wave, and combining their predictions into a final label [23]. The combination can be done using various methods within a voting or stacking strategy. This approach is called SepWav ("Separate Waves") in [2].

The aggregation function and MerWavTime(–) approaches are intuitive and widely utilised in the literature. The SepWav approach is more elaborated and has been utilised in some research (e.g., [23]); however, some of the ways for merging the classifiers' predicted labels listed above can be considered a novel contribution of this work.

Step 2 – Feature Selection: Based on Step 1's result, Auto-Sklong either selects a feature selection method or proceeds without it. In the data transformation approach, the system randomly decides whether to include the standard Correlation-based Feature Selection (CFS) method [26]. In the algorithm adaptation approach, it randomly decides whether to include "Exhaustive CFS per Group" (Exh-CFS-perGr), a longitudinal variation of CFS [21], [27].

If Exh-CFS-perGr is chosen, its hyperparameter "phase" is set to either "Phase 1 only" or "Phase 1 and 2" (default). "Phase 1 only" applies CFS to temporal variations of each feature to select the best ones, merging them into a single set. "Phases 1 and 2" performs Phase 1 and then applies standard CFS to the selected features plus any non-longitudinal features.

Step 3 – Classifier Selection: Auto-Sklong randomly chooses a classification algorithm (among candidate algorithms resulting from Step 1) and randomly sets its hyperparameters.

Table III lists the candidate $traditional$ algorithms and their hyperparameter settings following the data transformation approach; Table IV lists the $longitudinal$ algorithms and their hyperparameters following the algorithm adaptation approach.

Among the 5 longitudinal algorithms, 4 ("Lexico Random Forest", "Lexico Decision Tree", "Lexico Deep Forest", and "Lexico Gradient Boosting") are decision tree-based methods that use lexicographic optimisation to favour features measured in recent waves, based on the principle that recent measurements are more predictive and acceptable to users (e.g., a recent cholesterol level is more relevant than an older one). "Lexico Random Forest" and "Lexico Decision Tree" were described in [10], while "Lexico Deep Forest" and "Lexico Gradient Boosting" are novel lexicographic versions of the deep forest [28] and Gradient Boosting [29] algorithms.

Auto-Sklong's search space also includes Nested Trees [30], a longitudinal algorithm where each node of an outer decision tree is an inner decision tree built from temporal variations of the same longitudinal feature.

A limitation of Auto-Sklong's current search space is the absence of deep neural network methods for temporal data [31]. We focus on decision tree ensembles because they are the state-of-the-art for tabular data, often outperforming deep learning [32], [33], and are generally much faster — an important factor in AutoML. In any case, the current search space has already yielded good results – see Section V.

B. Search Strategy and Implementation Details

Auto-Sklong (Scikit-Learn API compliant [8]) supports 4 search methods: BO, Evolutionary Algorithms, Successive Halving, and Random Search [5]; the latter 3 were implemented via the General Automated Machine Learning Assistant (GAMA) [34]. In this work, we used Auto-Sklong with its default BO method with Expected Improvement as the acquisition function [5]. BO was implemented within GAMA, enhanced with SMAC3 [35].

Auto-Sklong can output classification pipelines containing both longitudinal and non-longitudinal methods. It achieves this by utilising our Longitudinal ML toolkit, *Sklong* (short for "Scikit-Longitudinal"), for longitudinal methods, and the Scikit-Learn library [8] for non-longitudinal methods.

Both *Auto-Sklong* and *Sklong* are open-source and available on GitHub: Auto-Sklong and Sklong.

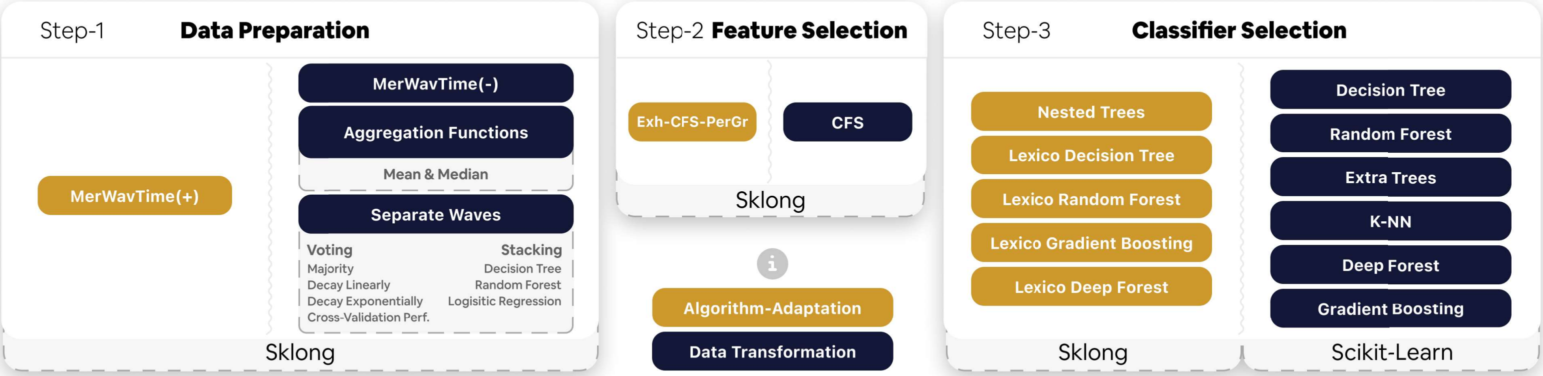

Fig. 1: **Auto-Sklong's Sequential Search Space**: The 3 steps to construct a classification pipeline. Gold-yellow (lighter shade) and blue-navy (darker shade) colours indicate longitudinal and non-longitudinal methods, respectively. The ML library used in each step is noted at the bottom of each box.

IV. Experimental Setup

A. Datasets Used in the Experiments

We used 20 longitudinal datasets from [36], based on the ELSA database [7], which tracks United Kingdom participants aged 50 or older through repeated interviews. The ELSA-Nurse datasets consist of biomedical data collected every four years by health professionals, while the ELSA-Core datasets come from core interviews conducted every two years. Each dataset combines either Nurse or Core features with one of 10 age-related diseases as the binary class variable from wave 8. The ELSA-Nurse datasets contain 7,096 instances and 140 features from waves 2, 4, 6, and 8. The ELSA-Core datasets have 8,405 instances and 171 features from waves 1–7 (plus age from wave 8). Details of dataset creation are in [36].

In the *diabetes* dataset with Nurse data, we removed the longitudinal feature "HbA1c" because it is directly used clinically to diagnose diabetes, making its use for prediction "unfair".

Table I presents the class distribution of each dataset, showing the percentage of positive instances (individuals with the disease) and the class imbalance ratio — the number of majority (negative) instances divided by minority (positive) instances — for both Nurse and Core datasets.

TABLE I: Class distribution in each dataset

Disease	Posit. class % (Nurse)	Posit. class % (Core)	Class imbalance ratio (Nurse)	Class imbalance ratio (Core)
Arthritis	42.57%	39.65%	1.35	1.52
Hbp	40.21%	38.72%	1.49	1.58
Cataract	32.72%	29.60%	2.06	2.38
Diabetes	13.33%	12.83%	6.50	6.80
Osteoporosis	9.22%	8.45%	9.85	10.84
Stroke	5.93%	5.45%	15.86	17.35
Heart attack	5.65%	5.25%	16.70	18.06
Angina	3.64%	3.39%	26.51	28.49
Dementia	2.09%	1.92%	46.95	51.20
Parkinsons	0.93%	0.89%	106.53	111.07

B. Evaluating Predictive Accuracy

Predictive accuracy was assessed using the well-known Area Under the ROC curve (AUROC) [13], which is also the optimisation measure for Auto-Sklong and Auto-Sklearn.

We used nested cross-validation to evaluate each AutoML system. The dataset was divided into 5 outer folds for the outer cross-validation. Each outer fold served once as the test set, with the remaining folds used for training. For each training set, an inner 5-fold cross-validation measured the AUROC of candidate solutions evaluated by the BO search. The reported AUROC results are the mean values over the 5 outer test sets. The two RF methods were evaluated using standard 5-fold cross-validation without inner cross-validation, and their AUROC results are also mean values over the 5 test sets.

C. The AutoML systems' and Baseline Methods' Settings

We compared Auto-Sklong against 3 other methods: (a) Standard Random Forest (RF) from Scikit-learn [8], a non-longitudinal baseline; (b) Lexicographic RF (implemented in *Sklong*), a longitudinal baseline [10]; (c) Auto-Sklearn [37], a state-of-the-art AutoML system for standard classification.

The first two baseline methods used default hyperparameter settings, as RF's defaults generally perform well [38]. Since Auto-Sklearn and standard RF do not include longitudinal methods, they were applied after transforming the longitudinal data into standard tabular format using the MerWavTime(–) approach (see Subsection III-A). Auto-Sklearn was used with default settings, including its meta-learning and post-hoc ensembling [39], which are not available in Auto-Sklong.

Both AutoML systems were given a runtime budget of 24 hours per run (i.e., per outer fold of the nested cross-validation) and a maximum evaluation time of 1,000 seconds per candidate solution during inner cross-validation. Each run was allocated one Intel Xeon E5520 CPU and 20 GB of RAM.

V. Computational Results and Discussion

Table II shows the AUROC results for 4 systems / methods: the proposed Auto-Sklong, Auto-Sklearn, and two baseline RF methods (standard and longitudinal). Results are shown across the 20 datasets described in Subsection IV-A — 10 with Nurse (N) data and 10 with Core (C) data from the ELSA study. For each dataset, the best result is highlighted in bold.

Auto-Sklong achieved the highest AUROC in 13 of the 20 datasets, while Auto-Sklearn was superior in the remaining 7 datasets (mainly Diabetes, HBP, and Cataract datasets).

We compared Auto-Sklong to each of the 3 other methods using a two-tailed Wilcoxon signed-rank statistical test [40] at the significance level $\alpha = 0.05$. This non-parametric test does not require the assumption of normality. The test results indicated that Auto-Sklong's AUROC values were significantly better than those of Auto-Sklearn (p = 0.025), standard RF (p<0.001), and lexicographic RF (p<0.001).

Interestingly, Auto-Sklong outperformed Auto-Sklearn despite Auto-Sklearn's advanced meta-learning and post-hoc ensembling procedures, which are not present in Auto-Sklong.

Meta-learning [39] provides a warm-start set of candidate solutions based on classifiers that performed well on similar datasets. However, since Auto-Sklearn's meta-learning relies on similarities with standard, non-longitudinal datasets, these warm-start solutions are not tailored for longitudinal classification — consistent with its non-longitudinal search space.

Also, Auto-Sklearn's post-hoc ensembling combines many high-performing pipelines found by the BO search, to enhance predictive accuracy. Again, this procedure is not designed for longitudinal classification, as it combines non-longitudinal classifiers. However, when Auto-Sklong returns a non-longitudinal pipeline as the best solution, Auto-Sklearn's use of ensembling should give it an edge over Auto-Sklong.

TABLE II: AUROC values on 20 datasets, for Auto-Sklong, Auto-Sklearn, and two baseline random forest (RF) methods

Dataset	Auto-Sklong	Auto-Sklearn	RF	Lexico-RF
Arthritis (N)	**0.6848**	0.675	0.6696	0.6726
Diabetes (N)	0.8822	**0.8846**	0.8702	0.8684
HBP (N)	0.7674	**0.7714**	0.7616	0.7572
Cataract (N)	0.7318	**0.735**	0.7198	0.7208
Osteoporosis (N)	**0.7452**	0.7434	0.7240	0.7226
Angina (N)	**0.7638**	0.6742	0.7228	0.7142
Dementia (N)	**0.8064**	0.6756	0.7562	0.7592
Heart Attack (N)	**0.793**	0.7118	0.7564	0.7534
Parkinsons (N)	**0.6454**	0.5146	0.5324	0.6044
Stroke (N)	**0.7536**	0.746	0.7266	0.7308
Arthritis (C)	0.8162	**0.8204**	0.8100	0.8068
Diabetes (C)	0.809	**0.8108**	0.7866	0.7850
HBP (C)	0.7164	**0.7184**	0.7012	0.7064
Cataract (C)	0.7524	**0.7654**	0.7424	0.7414
Osteoporosis (C)	**0.7848**	0.773	0.7410	0.7478
Angina (C)	**0.8124**	0.788	0.7532	0.7584
Dementia (C)	**0.8966**	0.8934	0.8482	0.8468
Heart Attack (C)	**0.7886**	0.7648	0.7342	0.7254
Parkinsons (C)	**0.7988**	0.7754	0.6868	0.7062
Stroke (C)	**0.7902**	0.7878	0.7652	0.7642

Notation: HBP = High Blood Pressure, RF = Random Forest, (N) and (C) denote the Nurse data and Core data in the ELSA database.

TABLE III: Hyperparameters for *traditional* classification algorithms – Data transformation approach

Algorithm	Hyperparameter	Candidate values	Default
Decision Tree	criterion	gini, entropy	gini
	max depth	2, 3, 4, 5, 6, 7, 8, 9, 10	2
	min samples split	$[2, 20]$	2
	min samples leaf	$[1, 20]$	1
Random Forest	criterion	gini, entropy	gini
	min samples split	$[2, 20]$	2
	min samples leaf	$[1, 20]$	1
	bootstrap	True, False	True
	n estimators	100, 150, 200, 250, 300, 350, 400, 450, 500, 1000	100
Extra Trees	criterion	gini, entropy	gini
	n estimators	100, 150, 200, 250, 300, 350, 400, 450, 500, 1000	100
	max features	$[0.0, 1.0]$	0.5
	min samples split	$[2, 20]$	2
	min samples leaf	$[1, 20]$	1
	bootstrap	True, False	False
KNN	n neighbors	1, 2, 3, 4, 5, 10, 15, 20, 30, 40, 50	1
	weights	uniform, distance	uniform
	p	1, 2	2
Deep Forest	n estimators	2, 3	2
Gradient Boosting	max depth	2, 3, 4, 5, 6, 7, 8, 9, 10	2
	learning rate	0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.2, 0.3, 0.4, 0.5	0.1
	n estimators	100, 150, 200, 250, 300, 350, 400, 450, 500, 1000	100

TABLE IV: Hyperparameters for *longitudinal* classification algorithms – Algorithm adaptation approach

Algorithm	Hyperparameter	Candidate values	Default
Nested Tree	max outer depth	5, 6, 7, 8, 9, 10	10
	max inner depth	2, 3, 4, 5	5
	min node size (for both outer and inner trees)	2, 3, 4, 5, 6, 7, 8, 9, 10	2
Lexico Decision Tree	threshold gain	0.0, 0.001, 0.0015, 0.002, 0.0025, 0.003, 0.0035, 0.004, 0.0045, 0.005, 0.01	0.02
	max depth	2, 3, 4, 5, 6, 7, 8, 9, 10	5
	min samples split	$[2, 20]$	2
	min samples leaf	$[1, 20]$	1
Lexico Random Forest	threshold gain	0.0, 0.001, 0.0015, 0.002, 0.0025, 0.003, 0.0035, 0.004, 0.0045, 0.005, 0.01	0.02
	min samples split	$[2, 20]$	2
	min samples leaf	$[1, 20]$	1
	bootstrap	True, False	True
	n estimators	100, 150, 200, 250, 300, 350, 400, 450, 500, 1000	100
Lexico Deep Forest	classifier type	LexicoRFClassifier, LexicoCompleteRFClassifier	LexicoRFClassifier
	n estimators	2, 3	2
Lexico Gradient Boosting	max depth	2, 3, 4, 5, 6, 7, 8, 9, 10	2
	learning rate	0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.2, 0.3, 0.4, 0.5	0.1
	n estimators	100, 150, 200, 250, 300, 350, 400, 450, 500, 1000	100
	threshold gain	0.0, 0.001, 0.0015, 0.002, 0.0025, 0.003, 0.0035, 0.004, 0.0045, 0.005, 0.01	0.0

VI. Conclusions

We proposed Auto-Sklong, a new AutoML system designed to solve the CASH problem for longitudinal classification. Auto-Sklong was evaluated on 20 datasets from the ELSA database, combining two types of features (Nurse and Core ELSA data) with 10 age-related diseases as class variables.

In these experiments, Auto-Sklong outperformed – with statistical significance – both a state-of-the-art AutoML system (Auto-Sklearn) and two versions of RF as baseline methods, one standard (non-longitudinal) and one longitudinal version.

Future research could extend Auto-Sklong's search space to include more longitudinal classification methods; or to include meta-learning and post-hoc ensembling techniques.

References

[1] E. K. Kelloway and L. Francis, "Longitudinal research and data analysis," in *Research methods in occupational health psychology*. Routledge, 2012, pp. 374–394.

[2] C. Ribeiro and A. A. Freitas, "A mini-survey of supervised machine learning approaches for coping with ageing-related longitudinal datasets," in *3rd Workshop on AI for Aging, Rehabilitation and Independent Assisted Living (ARIAL), part of IJCAI-19*, 2019, 5 pages.

[3] A. Allam, S. Feuerriegel, M. Rebhan, M. Krauthammer *et al.*, "Analyzing patient trajectories with artificial intelligence," *Journal of medical internet research*, vol. 23, no. 12, p. e29812, 2021.

[4] A. Cascarano, J. Mur-Petit, J. Hernandez-Gonzalez, M. Camacho, N. de Toro Eadie, P. Gkontra, M. Chadeau-Hyam, J. Vitria, and K. Lekadir, "Machine and deep learning for longitudinal biomedical data: a review of methods and applications," *Artificial Intelligence Review*, vol. 56, no. Suppl 2, pp. 1711–1771, 2023.

[5] M. Baratchi, C. Wang, S. Limmer, J. N. van Rijn, H. Hoos, T. Bäck, and M. Olhofer, "Automated machine learning: past, present and future," *Artificial Intelligence Review*, vol. 57, no. 5, Apr. 2024.

[6] J. Waring, C. Lindvall, and R. Umeton, "Automated machine learning: Review of the state-of-the-art and opportunities for healthcare," *Artificial intelligence in medicine*, vol. 104, p. 101822, 2020.

[7] S. Clemens, A. Phelps, Z. Oldfield, M. Blake, A. Oskala, M. Marmot, N. Rogers, J. Banks, A. Steptoe, and J. Nazroo, "English longitudinal study of ageing: Waves 0-8, 1998-2017," 2019. [Online]. Available: https://beta.ukdataservice.ac.uk/datacatalogue/doi/?id=5050#16

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[9] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

[10] C. Ribeiro and A. A. Freitas, "A lexicographic optimisation approach to promote more recent features on longitudinal decision-tree-based classifiers: applications to the english longitudinal study of ageing," *Artificial Intelligence Review*, vol. 57, no. 4, Mar. 2024.

[11] C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Auto-weka: Combined selection and hyperparameter optimization of classification algorithms," in *Proc. of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013, pp. 847–855.

[12] A. Sinha, P. Malo, and K. Deb, "A review on bilevel optimization: From classical to evolutionary approaches and applications," *IEEE Trans. on Evolutionary Computation*, vol. 22, no. 2, pp. 276–295, 2017.

[13] A. P. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," *Pattern recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.

[14] E. Brochu, V. M. Cora, and N. De Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," *arXiv preprint arXiv:1012.2599*, 2010.

[15] M. Feurer and F. Hutter, "Hyperparameter optimization," in *Automated machine learning*. Springer, Cham, 2019, pp. 3–33.

[16] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global optimization*, vol. 13, no. 4, pp. 455–492, 1998.

[17] Y. Zhang, D. Jarrett, and M. van der Schaar, "Stepwise model selection for sequence prediction via deep kernel learning," in *Proc. of the 23rd International Conference on Artificial Intelligence and Statistics, PMLR*, vol. 108, Aug 2020, pp. 2304–2314.

[18] O. Shchur, A. C. Turkmen, N. Erickson, H. Shen, A. Shirkov, T. Hu, and B. Wang, "Autogluon–timeseries: Automl for probabilistic time series forecasting," in *International Conference on Automated Machine Learning*. PMLR, 2023, pp. 9–1.

[19] W. Du, H. Cheung, C. A. Johnson, I. Goldberg, M. Thambisetty, and K. Becker, "A longitudinal support vector regression for prediction of als score," in *2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2015, pp. 1586–1590.

[20] L. Huang, Y. Jin, Y. Gao, K.-H. Thung, D. Shen, A. D. N. Initiative *et al.*, "Longitudinal clinical score prediction in alzheimer's disease with soft-split sparse regression based random forest," *Neurobiology of aging*, vol. 46, pp. 180–191, 2016.

[21] T. Pomsuwan and A. A. Freitas, "Feature selection for the classification of longitudinal human ageing data," in *2017 IEEE Internat. Conf. on Data Mining Workshops (ICDMW)*. IEEE, 2017, pp. 739–746.

[22] U. Niemann, T. Hielscher, M. Spiliopoulou, H. Völzke, and J.-P. Kühn, "Can we classify the participants of a longitudinal epidemiological study from their previous evolution?" in *2015 IEEE 28th Internat. Symp. on Computer-Based Medical Systems*. IEEE, 2015, pp. 121–126.

[23] S. Minhas, A. Khanum, F. Riaz, A. Alvi, S. A. Khan, A. D. N. Initiative *et al.*, "Early alzheimer's disease prediction in machine learning setup: Empirical analysis with missing value computation," in *Internat. Conf. on Intelligent Data Engineering and Automated Learning*. Springer, 2015, pp. 424–432.

[24] Y. Zhang, H. Jia, A. Li, J. Liu, and H. Li, "Study on prediction of activities of daily living of the aged people based on longitudinal data," *Procedia computer science*, vol. 91, pp. 470–477, 2016.

[25] J. Mo, S. Siddiqui, S. Maudsley, H. Cheung, B. Martin, and C. A. Johnson, "Classification of alzheimer diagnosis from adni plasma biomarker data," in *Proceedings of the International Conference on Bioinformatics, Computational Biology and Biomedical Informatics*, 2013, pp. 569–574.

[26] M. A. Hall, "Correlation-based feature selection for machine learning," Ph.D. dissertation, The University of Waikato, 1999.

[27] T. Pomsuwan and A. Freitas, "Feature selection for the classification of longitudinal human ageing data," Master's thesis, University of Kent, Feb 2018.

[28] Z.-H. Zhou and J. Feng, "Deep forest," *National Science Review*, vol. 6, no. 1, p. 74–86, Oct. 2018.

[29] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[30] S. Ovchinnik, F. Otero, and A. A. Freitas, "Nested trees for longitudinal classification," in *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, 2022, pp. 441–444.

[31] S. F. Ahmed, M. S. B. Alam, M. Hassan, M. R. Rozbu, T. Ishtiak, N. Rafa, M. Mofijur, A. Shawkat Ali, and A. H. Gandomi, "Deep learning modelling techniques: current progress, applications, advantages, and challenges," *Artificial Intelligence Review*, vol. 56, no. 11, pp. 13 521–13 617, 2023.

[32] L. Grinsztajn, E. Oyallon, and G. Varoquaux, "Why do tree-based models still outperform deep learning on typical tabular data?" *Advances in neural information processing systems*, vol. 35, pp. 507–520, 2022.

[33] R. Shwartz-Ziv and A. Armon, "Tabular data: Deep learning is not all you need," *Information Fusion*, vol. 81, pp. 84–90, 2022.

[34] P. Gijsbers and J. Vanschoren, "Gama: A general automated machine learning assistant," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2020, pp. 560–564.

[35] M. Lindauer, K. Eggensperger, M. Feurer, A. Biedenkapp, D. Deng, C. Benjamins, T. Ruhkopf, R. Sass, and F. Hutter, "Smac3: A versatile bayesian optimization package for hyperparameter optimization," *Journal of Machine Learning Research*, vol. 23, no. 54, pp. 1–9, 2022.

[36] C. E. Ribeiro, "New longitudinal classification approaches and applications to age-related disease data," Ph.D. Thesis, University of Kent, 2022.

[37] M. Feurer, A. Klein, K. Eggensperger, J. T. Springenberg, M. Blum, and F. Hutter, *Auto-sklearn: Efficient and Robust Automated Machine Learning*. Springer, 2019, pp. 113–134.

[38] P. Probst, M. N. Wright, and A. Boulesteix, "Hyperparameters and tuning strategies for random forest," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 9, 2018.

[39] M. Feurer, K. Eggensperger, S. Falkner, M. Lindauer, and F. Hutter, "Auto-sklearn 2.0: Hands-free automl via meta-learning," *Journal of Machine Learning Research*, vol. 23, no. 261, pp. 1–61, 2022.

[40] F. Wilcoxon, "Individual comparisons by ranking methods," in *Breakthroughs in statistics: Methodology and distribution*. Springer, 1992, pp. 196–202.