IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 15, No. 6, December 2026, pp. 5418∼5434
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i6.pp5418-5434

Meta-feature based algorithm selection on healthcare data classification

Mohamed Abdelbaki[1], Adnane Latif[1], Charafeddine Ait Zaouiat[2]
[1]Laboratory of Complex Systems Modeling, National School of Applied Sciences of Marrakech, Cadi Ayyad University, Marrakech, Morocco
[2]Department of Computer Science, Multidisciplinary Faculty of Sidi Bennour, Chouaib Doukkali University, El Jadida, Morocco

Article Info

Article history:

Received Dec 18, 2024
Revised Aug 9, 2026
Accepted Aug 31, 2026

Keywords:

Automated algorithm selection
Clinical decision support
Healthcare data classification
Medical informatics
Meta-learning

ABSTRACT

Healthcare organizations increasingly rely on classification models, but choosing a strong algorithm remains difficult for teams without deep machine learning expertise. This work studies offline, meta-feature-driven automated algorithm selection (AAS) for tabular healthcare data: this study extract statistical, information-theoretic, and landmarking descriptors, predict classifier performance with trained regressors, and rank eighteen diverse models under stratified nested cross-validation and leave-one-dataset-out (LODO) testing on thirty public healthcare benchmarks. The design favors transparency and low deployment cost over expensive online search across algorithms and hyperparameters. In ranking evaluation, the learning-to-rank (LTR) recommender achieves NDCG@1 of 0.978 for accuracy and 0.952 for F1-score, indicating that compact meta-features steer users toward near-optimal choices across heterogeneous tasks. This study contribute a reproducible benchmark-oriented pipeline and a practical blueprint for integrating offline algorithm selection into clinical analytics workflows.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Mohamed Abdelbaki
Laboratory of Complex Systems Modeling, National School of Applied Sciences of Marrakech
Cadi Ayyad University
Marrakech, Morocco
Email: abdelbaki.mo6@gmail.com

1. INTRODUCTION

Healthcare data classification supports diagnosis, prognosis, and operational analytics, but choosing a strong classifier remains difficult for teams without deep machine learning expertise. Automated algorithm selection (AAS) addresses this challenge by recommending algorithms from measurable dataset properties. Following Rice's algorithm-selection framework [1] and the meta-learning literature [2]–[4], this study examines a meta-feature-driven selector that operates offline on tabular healthcare data; section 2 reviews related work in more detail.

Healthcare teams often lack dedicated modeling expertise [5]–[7], while online automatic machine learning (AutoML) frameworks that search many pipelines at runtime can be computationally expensive and difficult to interpret in clinical settings [8]–[10]. Recent surveys and empirical studies emphasize diverse problem portfolios, effective meta-features, and explainable recommendation as prerequisites for trustworthy selection [11]–[14]. Complementary lines of work have advanced offline meta-learning recommenders for industrial and manufacturing settings [15]–[17], ensemble recommendation [18], feature-engineering

guidance [19], and clinical or biomedical AutoML case studies [20]–[23].

The novelty of this work lies in bringing that offline meta-learning paradigm to a curated healthcare classification portfolio with an explicit emphasis on interpretability rather than runtime pipeline search. This study make two contributions tailored to this gap. First, compile a reproducible benchmark of 30 public healthcare classification datasets for algorithm-selection research. Second, implement an offline pipeline that extracts statistical, information-theoretic, and landmarking meta-features, predicts classifier performance with trained regressors, and ranks 18 models under stratified nested cross-validation and leave-one-dataset-out (LODO) evaluation, offering lower deployment cost and greater transparency than exhaustive online search.

The positioning differs from online AutoML systems such as AutoPrognosis [8], Auto-sklearn [24], and automatic disease classification (AutoDC) [9], which search pipelines and hyperparameters at runtime and can achieve strong accuracy at substantial computational cost. This study instead pre-train meta-regressors on a healthcare benchmark and recommend classifiers from interpretable dataset descriptors without re-running large search jobs for each new dataset. This design targets three gaps in clinical analytics: a curated 30-dataset healthcare portfolio for algorithm-selection research, transparent Python meta-feature extractor (PyMFE) meta-features linked to algorithm behavior (section 4.5), and decision-support recommendations rather than replacement of end-to-end AutoML pipelines. Universal superiority over search-based AutoML on all tasks is not claimed; instead, exhaustive runtime optimization is traded for efficiency, explainability, and lower deployment overhead in resource-constrained hospital settings.

The rest of this paper is organized as follows. Section 2 explains the meta-learning framework and how the problem, algorithm, feature, and performance spaces interact in our setting. Section 3 lists the used datasets and the evaluation protocol, including the dataset registry, preprocessing, the model zoo, and nested validation. Section 4 presents results, interprets findings, and puts forward some practical implications for clinical analytics workflows. Section 5 presents a short conclusions, then states limitations and future work.

2. MATERIALS AND METHODS

2.1. Algorithm selection and machine learning automation

The problem of algorithm selection can be considered a multidimensional model defined by four spaces which interact with each other. The problem space defines the set of problems or instances which can be solved whose specific characteristics can influence the choice of the algorithm and its performance. The algorithm space is the set of algorithms applicable to the space of problems and is where the selection happens.

The feature space groups measurable properties which can describe instances, they should allow to distinguish between the different problems and anticipate algorithm performance. Finally, performance measurement space is the set of measurements used to quantify the performance of algorithm on a given instance [1]. Figure 1 summarizes how these spaces interact in Rice's formulation. This framework is instantiated for supervised tabular healthcare classification.

2.2. The evolution of automated algorithm selection strategies for classification problems

The problem space is framed as tabular classification tasks: instances are rows, features and labels define the mapping to be learned [25], and the algorithm space comprises candidate classifiers trained to generalize to new examples. In this study those tasks are healthcare tables, so the instances correspond to patients or cases rather than images or free text. Performance is judged against standard classification metrics [26]; operational definitions and healthcare-oriented measures used in benchmark are specified in section 3.

AAS has evolved from early systems such as EUREKA [27] toward per-instance recommendation [3]. Landmarking and meta-dataset benchmarks from the StatLog project [28] established dataset featurization as a practical substrate for selection; portfolio methods such as SATzilla [29] demonstrated strong gains on structured problem classes. More recently, offline meta-learning tools (e.g., AMLBID [15]) and nearest-neighbor recommendation in meta-feature space [17] deliver low-latency guidance without re-running large search procedures at deployment time. Portfolio configuration frameworks such as AutoFolio extend this line by combining algorithm portfolios with automated per-instance tuning [30], [31]. Meta-learning has also been applied to recommend classifier ensembles rather than single models [18].

Clinical evaluations of end-to-end AutoML tools report competitive but search-intensive performance on tabular diagnosis tasks [10]. As outlined in section 1, such online pipelines remain common in healthcare deployments, yet curated benchmarks for offline, interpretable algorithm recommendation in this

domain remain scarce; the contrast with runtime AutoML frameworks summarized there motivates the healthcare-specific offline design. This work targets that gap by benchmarking meta-feature-driven selection on 30 public healthcare classification tasks. The following subsection formalizes dataset representation, the meta-dataset, and regression-based selection; extraction and evaluation protocols are specified in section 3.

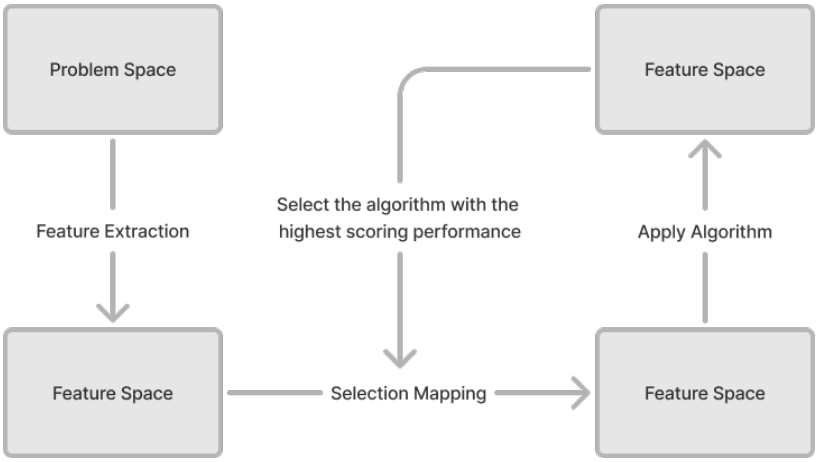

Figure 1. The John R. Rice's algorithm selection model component interactions

2.3. Data representation and regression-based selection

Each dataset is represented as a feature-label pair D in (1). For p datasets, featurization yields a vector of q meta-features per dataset, assembled as the meta-dataset matrix MtD in (2). Empirical benchmark scores for m algorithms populate the performance matrix P in (3) (18 algorithms, 30 datasets; section 3). For each algorithm j, training rows concatenate MtD with column j of P, forming the augmented matrix Mtd_j in (4).

$$D = (\vec{f_{ij}}) + (c_i) \tag{1}$$

$$MtD = \begin{bmatrix} mf_{1,1} & mf_{1,2} & \cdots & mf_{1,q} \\ mf_{2,1} & mf_{2,2} & \cdots & mf_{2,q} \\ \vdots & \vdots & \ddots & \vdots \\ mf_{p,1} & mf_{p,2} & \cdots & mf_{p,q} \end{bmatrix} \tag{2}$$

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,m} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ p_{p,1} & p_{p,2} & \cdots & p_{p,m} \end{bmatrix} \tag{3}$$

$$Mtd_j = MtD + P(:,j) = \begin{bmatrix} mf_{1,1} & \cdots & mf_{1,q} & p_{1,j} \\ mf_{2,1} & \cdots & mf_{2,q} & p_{2,j} \\ \vdots & \ddots & \vdots & \vdots \\ mf_{p,1} & \cdots & mf_{p,q} & p_{p,j} \end{bmatrix} \tag{4}$$

This study adopt a regression-based framework to approximate the performance $p(a_i, F_j)$ of algorithm a_i on feature configuration F_j (5). To ensure cross-algorithm comparability during meta-modeling, all performance metrics are normalized to the $[0, 1]$ range (metric definitions in section 3). The recommended classifier maximizes predicted performance (6); predicted scores induce the algorithm ranking evaluated in section 4.

$$p'(a_i, F_j) \approx p(a_i, F_j) \quad \forall j \in 1, ..., n, \forall a_i \in A \tag{5}$$

$$\max_{a_i \in A} p'(a_i, F_j) \tag{6}$$

At deployment, an uploaded dataset yields a meta-feature row in MtD; (5) predicts each algorithm's score and (6) selects the best—the practical output of the recommender before list-wise ordering in subsection 4.6.

3. EXPERIMENTAL SETUP: METHOD

This section specifies datasets, preprocessing, meta-feature extraction, the evaluated model zoo, and the performance evaluation protocol used to populate matrix P in (3). Those specifications are held fixed across algorithms so that later comparisons, and the selector trained on P, rest on the same benchmark. Figure 2 summarizes this experiment end-to-end.

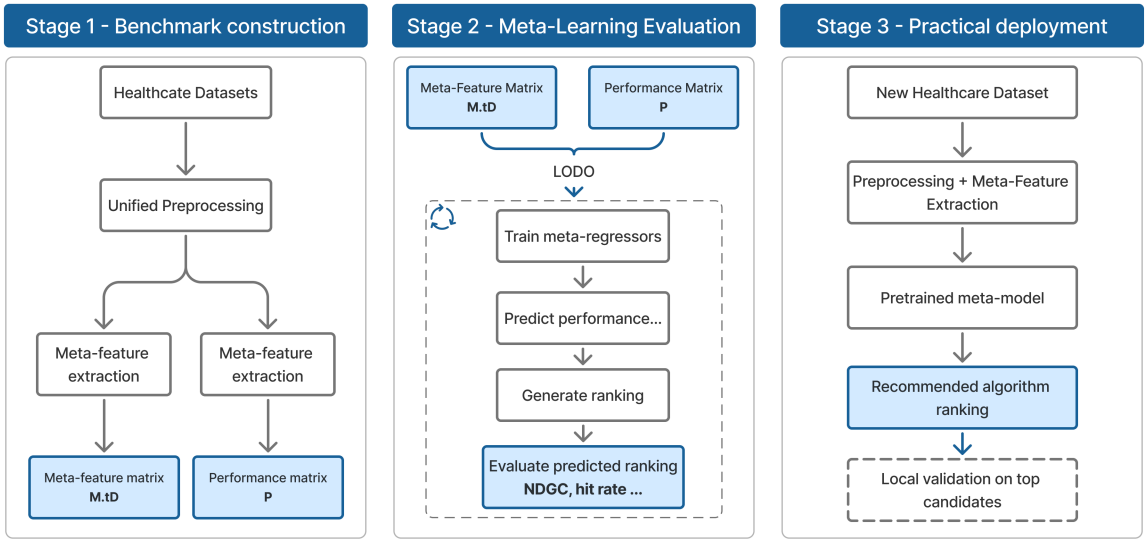

Figure 2. Overview of the experimental setup

3.1. Dataset collection and registry

A total of 30 publicly available healthcare-related tabular classification datasets were assembled, spanning cardiovascular, endocrine, oncology, infectious-disease, obstetric, and other clinical domains. Instance counts range from 58 (preterm) to 1,048,575 (covid19), with 13 features on average (range: 6–188) and predominantly binary targets. Table 1 lists each dataset's source, sample size, feature and class counts, documented geography, and demographic coverage as reported in the original source descriptions.

A centralized registry records the dataset name, file path, target variable, source URL(s), and any dataset-specific edits applied before our unified pipeline. The 30 sources differ in file format, target-column names, and earlier notebook cleaning, so this record is what lets one preprocessing path load every file and remain reproducible if a source is updated. Where primary sources document geography or demographics, these are summarized in Table 1; otherwise this study state "Not reported in source".

Several datasets carry documented geographic or population concentration: Mexican national COVID-19 surveillance (covid19), a female Pima Indian cohort in Arizona (diabetes), the Framingham Heart Study in Massachusetts (framingham), North-East Andhra Pradesh liver patients (indianLiver), and a Tehran university preterm cohort with only 58 cases (preterm). The heartDisease file merges five legacy UCI heart-disease cohorts (USA, Hungary, Switzerland). Other entries lack stated geography (cardioDisease, obesity) or are not patient-level records: symptoms is a web-compiled symptom–disease knowledge base, and stress contains literature-derived SaYoPillow values with no human subjects. The appendix cites primary upstream sources (UCI, PhysioNet, government open data, or dataset papers); Kaggle-hosted files were used as download mirrors where needed, and any third-party notebook preprocessing is documented in the experiment registry.

Findings therefore apply to the specific sources and populations documented in Table 1. Some datasets did not provide geographical or demographics data as explained on the table. Table 1 provides the complete registry of 30 datasets used in this study. The source column lists the primary upstream repository or

publication; Kaggle mirrors were used only as download paths where applicable. Full URLs and preprocessing notes are in the experiment registry (index.json).

Table 1. Complete dataset registry ($N = 30$)

Dataset	Domain	n	Feat.	Cl.	Source	Geography	Demographics
Covid19	Infectious disease (COVID-19)	1,048,575	20	2	Mexico COVID-19 open data (gob.mx)	Mexico (Mexican government surveillance)	Age, sex (Mexican surveillance schema)
Diabetes	Endocrine / diabetes	768	8	2	UCI Pima Indians Diabetes (NIDDK)	Arizona USA (Pima Indian community)	Female-only cohort age 21+; Pima Indian heritage (NIDDK)
Symptoms	Multi-disease symptom mapping	4,920	17	41	Web-compiled symptom–disease KB (Kaggle)	Not reported in source	Not reported in source
Stroke	Cerebrovascular	5,110	10	2	McKinsey stroke dataset (Bangladesh)	Bangladesh (secondary literature; primary confidential)	Age, gender in feature set
HepatitisC	Hepatology	589	12	5	Hoffmann et al. (2018)	Not reported in source	Not reported in source
Cirrhosis	Hepatology	276	16	2	UCI Cirrhosis Patient Survival	Not reported in source	Age, sex on UCI variable list
HeartDisease	Cardiovascular	918	11	2	UCI Heart Disease (5-cohort merge)	Multi-site: USA, Hungary, Switzerland (5-dataset UCI merge)	Age, sex (UCI merge attributes)
HeartFailure	Cardiology / heart failure	299	12	2	Chicco & Jurman (2020) BMC	Not reported in source	Age, sex (BMC paper / UCI)
CardioDisease	Cardiovascular screening	66,488	12	2	Kaggle cardiovascular screening cohort	Not reported in source	Age, gender (age in days)
IndianLiver	Hepatology	570	8	2	UCI ILPD	North-East Andhra Pradesh India	Age, gender (UCI counts)
HeartCleaveland	Cardiovascular	293	14	2	UCI Cleveland Heart Disease	Cleveland Ohio USA	Age, sex (UCI Cleveland subset)
Wisconsin	Oncology / breast	569	31	2	UCI Wisconsin Breast Cancer Diagnostic	Wisconsin USA	Not reported in source
Hypothyroid	Endocrine / thyroid	3,711	30	2	UCI Thyroid Disease	Not reported in source	Verify age/sex in 30-column subset
Appendicitis	General surgery	106	7	2	UCI Appendicitis	Not reported in source	Not reported in source
Framingham	Cardiovascular epidemiology	3,656	16	2	Framingham Heart Study	Framingham Massachusetts USA	Sex, age; 10-year CHD risk (Framingham Heart Study)
Lowerback	Musculoskeletal / spine	310	13	2	UCI Vertebral Column	Not reported in source	Not reported in source
BreastCancer	Oncology / breast	569	31	2	UCI Wisconsin Breast Cancer Diagnostic	Wisconsin USA	Not reported in source
Heart2020	Cardiovascular epidemiology	319,795	13	2	CDC 2020 BRFSS	United States (CDC 2020 BRFSS)	Age, sex in survey; subset may omit race/ethnicity
Autism	Neurodevelopment screening	609	16	2	UCI Autism Screening Adult	Not reported in source	Screening instrument may include age
Obesity	Obesity classification	108	6	4	Kaggle obesity classification	Not reported in source	Age, gender in columns
HeartBeat	Cardiology / ECG	14,552	188	2	PhysioNet MIT-BIH Arrhythmia	Boston USA (MIT-BIH / PhysioNet)	Not reported in source
Stress	Sleep / stress physiology	630	9	5	SaYoPillow (literature synthetic)	Not applicable (synthetic)	Not reported in source
FetalHealth	Obstetrics / fetal monitoring	2,126	22	3	UCI Cardiotocography	Not reported in source	Not reported in source
FetalCardio	Obstetrics / fetal monitoring	2,126	10	3	UCI Cardiotocography	Not reported in source	Not reported in source
Diabetes2019	Endocrine / diabetes	952	30	2	Tigga & Garg (2020) Procedia CS	Not reported in source	Check Tigga & Garg paper for age/sex
Thoracic	Thoracic surgery / oncology	470	29	2	UCI Thoracic Surgery	Not reported in source	Check UCI/paper for age/sex
Preterm	Obstetrics	58	6	2	Tehran EHG preterm study	Tehran Iran (University of Science and Research)	Maternal EHG signals; n=58
ArrhythmiaBinary	Cardiology / ECG	100,689	33	2	PhysioNet MIT-BIH Arrhythmia	Boston USA (MIT-BIH derived)	Not reported in source
CervicalCancer	Gynecologic oncology	858	34	2	UCI Cervical Cancer Risk Factors	Not reported in source	Age, behavioral factors (UCI)
Epilepsy	Neurology / EEG	11,500	178	2	UCI Epileptic Seizure Recognition	Not reported in source	Not reported in source

3.2. Meta-feature extraction

Using PyMFE, this study extracted the full set of meta-features for each dataset [14]. These features allow us to have a statistical and structural description of the data. This study relied on the complimentary nature of these features bundled into groups to have a solid description of the datasets: general measures such as number of instances, attributes, and classes per dataset. Statistical measures including mean, variance, kurtosis and variance across features. Information theoretic measures like entropy and mutual information to assess the information content of variables and their redundancy. And landmarking measures which provide a first glance at the performance of algorithms based on simple predictors [12]. The choice of these categories plays a role in the interpretability of the system. The use of interpretable statistical and information theoretic measures makes it easier to identify data properties more precisely, which can advantage or disadvantage certain families of predictors such as high sparsity or low class entropy. This study explore these observations in further detail in subsection 4.5. This transparency is particularly important in the clinical domain where recommendations that are difficult to explain can be on of no use to practicians.

PyMFE yields approximately 180 attributes per dataset. Rather than applying a fixed upfront cap (e.g., retaining only the top K features per performance metric via univariate filters), the full set of numerically stable descriptors is retained after quality control: columns with missing values or zero variance are removed, and correlation analyses additionally exclude features with insufficient coverage across the 30-dataset benchmark. With $p \gg n$ at the meta level, complexity is controlled at modelling time through regularized regressors (Lasso, ElasticNet, Ridge) trained under LODO validation; sparsity is therefore induced by the chosen penalty and nested cross-validation rather than by a hard pre-selection step. This preserves named, clinically interpretable descriptors for the correlation analysis in subsection 4.5. Alternative dimensionality-reduction strategies such as principal component analysis (PCA) or autoencoder-based encodings can compress meta-feature space and have improved nearest-neighbor recommendation in related work [16], but they yield latent components that are harder to explain to domain experts. The approach therefore favor transparent statistical and landmarking measures combined with regularized meta-regression over unsupervised compression, accepting some redundancy among correlated PyMFE groups in exchange for explainability.

3.3. Unified preprocessing pipeline

To address the variability in data quality across healthcare sources, this study implemented a unified preprocessing pipeline. Unlike earlier versions that relied on row removal, the current framework imputes missing numerical values using the median and missing categorical values using the most-frequent category, then standardizes numerical features via Z-score scaling and encodes categorical variables using one-hot encoding. This ensures that all downstream classification algorithms receive a clean, scaled, and consistent input matrix X. In parallel, meta-learning has also been explored to recommend preprocessing components (e.g., feature subset selection) based on dataset characterizations, which motivates future extensions of the pipeline automation beyond classifier selection [19].

3.4. The evaluated model zoo

To build a broad and methodologically diverse meta-learning baseline, this study evaluated a zoo of 18 classification algorithms spanning distinct modeling paradigms. The reasoning behind our algorithm selection was to cover the most relevant and used algorithms for classification tasks on tabular data. The set included linear methods (Ridge classifier, logistic regression, stochastic gradient descent (SGD) classifier, perceptron, passive aggressive classifier), instance-based learners (k-nearest neighbors, nearest centroid), tree-based learners and ensemble methods (decision tree, random forest, gradient boosting, bagging, AdaBoost), and used linear support vector classifier (Linear SVC) to represent support vector machines as well as Gaussian naive Bayes as a probabilistic baseline predictor. To address the need for neural representations while keeping computation feasible, this study evaluated two multi-layer perceptron (MLP) variants: a standard feed-forward MLP with a fixed architecture, and a NAS-style MLP trained within an expanded architecture space (varying layers and hidden units) to emulate neural architecture search principles. The expanded search is used only while training that second network. At deployment, the offline selector therefore does not run a full neural architecture search for each new healthcare table. For readability, some figures use short algorithm abbreviations on axes; Table 2 lists the abbreviation used for each algorithm. This study deliberately focused on these 18 algorithms to ensure that the meta-learning task (predicting 18 target values) remains manageable given the 30-dataset meta-training set. At the same time, the zoo still provides a diverse enough menu for the AAS framework to choose from.

Table 2. Algorithm abbreviations used in figure axes

Abbrev.	Algorithm
Ada	AdaBoost
Bag	Bagging
DT	Decision tree
GBC	Gradient boosting
GNB	Gaussian naive Bayes
KNN	K-nearest neighbors
LDA	Linear discriminant analysis
LR	Logistic regression
LSVC	Linear SVC
MLP	Multi-layer perceptron
NAS	NAS-style MLP
NC	Nearest centroid
PAC	Passive-aggressive classifier
Perc	Perceptron
QDA	Quadratic discriminant analysis
RF	Random forest
Ridge	Ridge classifier
SGD	SGD classifier

3.5. Optimized performance evaluation pipeline

The evaluation process was designed to limit effect of imbalance between the classes, which is often present in healthcare datasets, and to strengthen the reliability of the results. Balanced class weighting was used during training to eliminate these imbalances, ensuring that a majority class cannot dominate the fit. Balanced accuracy was also reported alongside standard accuracy to show the difference with and without weighting.

This study also used a randomized search strategy with a fixed budget of 10 sampling iterations per model to ensure transparency and reproducibility in our hyperparameter optimization (HPO). A shared budget of this size prevents any one algorithm from receiving a much larger tuning effort than the others. The search spaces were defined using reasonable bounds for each algorithm, as summarized in Table 3. Continuous distributions are denoted by $U(min, max)$, while categorical options are listed within brackets. Examples include $C \in [0.1, 10]$ for linear models, $n_estimators \in [100, 500]$ for ensembles.

Table 3. HPO search spaces for the evaluated algorithm zoo

Algorithm family	Hyperparameter	Search space/range
Ridge classifier	alpha	$U(0.1, 10.0)$
Logistic regression	C	$U(0.1, 10.0)$
	solver	liblinear vs saga
Linear SVC	C	$U(0.1, 10.0)$
	loss	hinge vs squared_hinge
Decision tree	max_depth	[1, 10]
	min_samples_split	[2, 10]
Random forest	n_estimators	[100, 500]
	max_depth	[3, 10]
	min_samples_leaf	[1, 5]
Gradient boosting	n_estimators	[100, 500]
	learning_rate	$U(0.01, 0.2)$
	max_depth	[3, 10]
Bagging	n_estimators	[10, 100]
	max_samples	$U(0.5, 1.0)$
MLP	hidden_layer_sizes	[(50,), (100,), (50, 50)]
	activation	[tanh, relu]
	alpha	$U(0.0001, 0.005)$
NAS-style MLP	layers	[1, 3]
	units per layer	[16, 256]
	activation	tanh vs relu vs logistic
	learning_rate	constant vs adaptive vs invscaling

Classifier scores are estimated with stratified nested cross-validation ($K = 5$): the outer loop reports unbiased performance and the inner loop performs HPO (Table 3). For each dataset, this study store mean performance metrics; Subsection 4.1 summarizes mean, median, and standard deviation across the 30 datasets.

Meta-regressors are validated separately under LODO: for each held-out dataset, hyperparameters are selected by 3-fold cross-validation on the remaining 29 datasets, and predictions on the held-out point contribute to out-of-sample R^2, mean absolute error (MAE), and mean squared error (MSE) (subsection 4.6). This two-level design limits leakage between dataset characterization and performance modelling. Beyond accuracy and F1-score, this study report macro-averaged recall as sensitivity, macro one-vs-rest specificity from the confusion matrix, receiver operating characteristic (ROC) area under the curve (AUC) (weighted one-vs-rest for multi-class tasks; positive-class probability for binary tasks), and precision recall (PR)-AUC for binary tasks only ($n = 24$ of 30 datasets). PR-AUC is not computed for multi-class datasets in the current pipeline. Subsection 4.1 reports sensitivity, specificity, and AUC alongside accuracy and F1. Non-parametric tests in subsection 4.2 assess whether observed classifier differences across datasets are statistically significant.

4. RESULTS AND DISCUSSION

4.1. Algorithm performance assessment

This study evaluate 18 algorithms on 30 healthcare datasets under the protocol in section 3. Table 4 summarizes accuracy, balanced accuracy, F1-score, sensitivity (macro recall), and specificity across datasets, where sensitivity is defined as macro-averaged recall. Top ensembles remain strongest on clinically relevant rates: bagging achieves mean F1 ≈ 0.84 and mean sensitivity ≈ 0.84, while random forest leads on sensitivity (≈ 0.86) and specificity (≈ 0.88). Linear baselines such as the ridge classifier trail on sensitivity (≈ 0.72), consistent with the imbalance audit in subsection 4.4.

Table 4. Summary statistics (mean/median/std) of performance metrics across 30 datasets

Algorithm	accuracy			balanced_accuracy			f1_score			Sensitivity			specificity		
	mean	median	std	mean	median	std	mean	median	std	mean	median	std	mean	median	std
AdaBoost	0.880	0.925	0.152	0.779	0.831	0.201	0.775	0.839	0.215	0.779	0.831	0.201	0.840	0.892	0.160
Bagging	0.925	0.957	0.082	0.840	0.894	0.167	0.838	0.898	0.171	0.840	0.894	0.167	0.856	0.939	0.166
Decision tree	0.850	0.885	0.130	0.841	0.875	0.137	0.792	0.829	0.169	0.841	0.875	0.137	0.874	0.911	0.121
Gradient boosting	0.916	0.951	0.091	0.831	0.895	0.170	0.831	0.894	0.173	0.831	0.895	0.170	0.850	0.941	0.167
Gaussian naive Bayes	0.793	0.861	0.203	0.778	0.781	0.142	0.725	0.726	0.206	0.778	0.781	0.142	0.806	0.824	0.150
K-nearest neighbors	0.908	0.932	0.085	0.792	0.823	0.183	0.801	0.837	0.178	0.792	0.823	0.183	0.815	0.879	0.174
Linear discriminant analysis	0.890	0.902	0.083	0.788	0.791	0.148	0.793	0.795	0.140	0.788	0.791	0.148	0.823	0.849	0.147
Logistic regression	0.854	0.861	0.123	0.833	0.842	0.136	0.790	0.790	0.162	0.833	0.842	0.136	0.855	0.870	0.133
Linear SVC	0.855	0.864	0.111	0.826	0.839	0.130	0.784	0.788	0.152	0.826	0.839	0.130	0.854	0.878	0.128
Multi-layer perceptron	0.912	0.941	0.086	0.822	0.844	0.161	0.825	0.846	0.160	0.822	0.844	0.161	0.841	0.896	0.158
NAS-style MLP	0.906	0.943	0.090	0.815	0.812	0.163	0.816	0.837	0.167	0.815	0.812	0.163	0.835	0.888	0.161
Nearest centroid	0.802	0.793	0.122	0.785	0.760	0.125	0.734	0.707	0.150	0.785	0.760	0.125	0.821	0.830	0.126
Passive-aggressive classifier	0.841	0.839	0.111	0.814	0.825	0.132	0.770	0.760	0.151	0.814	0.825	0.132	0.847	0.875	0.130
Perceptron	0.837	0.840	0.119	0.807	0.821	0.130	0.762	0.744	0.152	0.807	0.821	0.130	0.842	0.862	0.133
Quadratic discriminant analysis	0.872	0.876	0.101	0.815	0.821	0.137	0.789	0.788	0.163	0.815	0.821	0.137	0.842	0.842	0.128
Random forest	0.897	0.921	0.100	0.860	0.899	0.141	0.834	0.878	0.158	0.860	0.899	0.141	0.880	0.924	0.127
Ridge classifier	0.872	0.905	0.098	0.719	0.723	0.192	0.722	0.747	0.194	0.719	0.723	0.192	0.770	0.781	0.181
SGD classifier	0.841	0.842	0.115	0.810	0.822	0.134	0.770	0.747	0.154	0.810	0.822	0.134	0.845	0.863	0.127

Table 5 reports ROC AUC over all 30 datasets and PR-AUC over the 24 binary classification tasks, as PR-AUC is not defined in the current pipeline for multi-class datasets. Top ensembles achieve mean ROC AUC above 0.91 (bagging ≈ 0.92) and mean PR-AUC above 0.78 on binary datasets (random forest ≈ 0.79, bagging ≈ 0.79). Friedman tests on sensitivity, specificity, and ROC AUC also reject equal performance across algorithms ($p \ll 0.001$). F1-score is considered the primary metric for algorithm selection and meta-learning ranking, while sensitivity, specificity, and AUC are also included for clinical completeness in subsection 4.6.

4.2. Statistical significance and robustness analysis

To assess whether performance differences among the 18 classification algorithms on the benchmark are statistically grounded, this study used Friedman and Wilcoxon ranking tests. Each test uses a dataset's nested CV performance score as one paired observation. In subsection 4.6, this study concluded support for the claims about relative classifier quality.

4.2.1. Overall difference: friedman test

We first ran the Friedman test, a non-parametric equivalent of the repeated-measures ANOVA, to test the null hypothesis that all algorithms perform equally across the 30 healthcare datasets. The test used three key metrics: F1-score, accuracy and balanced accuracy. As summarized in Table 6, the Friedman test rejected the null hypothesis for all three metrics ($p < 0.001$), confirming statistical significance in differences between the algorithms. This spread makes a single fixed default unsafe and motivates dataset-aware selection.

Table 5. AUC summary across algorithms

Algorithm	ROC AUC			PR-AUC (binary)		
	mean	median	std	mean	median	std
AdaBoost	0.904	0.940	0.115	0.775	0.864	0.266
Bagging	0.919	0.988	0.112	0.785	0.923	0.276
Decision tree	0.897	0.936	0.106	0.735	0.825	0.265
Gradient boosting	0.912	0.980	0.126	0.775	0.892	0.280
Gaussian naive Bayes	0.880	0.884	0.110	0.712	0.781	0.279
K-nearest neighbors	0.884	0.937	0.127	0.747	0.893	0.299
Linear discriminant Analysis	0.886	0.924	0.123	0.739	0.805	0.268
Logistic regression	0.883	0.918	0.130	0.737	0.792	0.269
Linear SVC	0.690	0.838	0.371	0.729	0.796	0.273
Multi-layer perceptron	0.906	0.946	0.109	0.768	0.877	0.273
NAS-style MLP	0.898	0.955	0.122	0.760	0.885	0.287
Nearest centroid	0.867	0.882	0.133	0.715	0.761	0.276
Passive-aggressive classifier	0.684	0.826	0.367	0.714	0.748	0.268
Perceptron	0.682	0.842	0.368	0.716	0.810	0.282
Quadratic discriminant analysis	0.900	0.927	0.098	0.752	0.828	0.266
Random forest	0.916	0.965	0.117	0.787	0.900	0.270
Ridge classifier	0.691	0.848	0.370	0.736	0.799	0.265
SGD classifier	0.877	0.914	0.124	0.720	0.790	0.284

Table 6. Friedman test results across algorithms (30 datasets)

Metric	Stat	p_value	n_datasets	n_algorithms
Accuracy_mean	188.9	4.754e-31	30	18
Balanced_accuracy_mean	92.75	1.927e-12	30	18
F1_score_mean	112.4	4.313e-16	30	18

4.2.2. Pairwise comparisons: wilcoxon signed-rank test

This study then proceeded with a post-hoc analysis using the Wilcoxon test to compare the top performing algorithm, identified as bagging classifier against all other candidates based on average ranking. F1-score is used as the primary metric due to its robustness under class imbalance. Table 7 presents the p-values as well as paired difference statistics of these comparisons. A Holm comparison has been applied for all comparisons to control family-wise error. After correction, the bagging classifier performed significantly better than several models and simpler baselines at a 95% confidence level ($p < 0.05$). Specifically, traditional linear models like the ridge classifier ($p = 2.11 \times 10^{-5}$), nearest centroid ($p = 8.35 \times 10^{-4}$), and Gaussian naive Bayes ($p = 1.32 \times 10^{-3}$) showed high performance gaps.

Performance differences between bagging and other strong ensembles such as random forest ($p = 0.587$) and gradient boosting classifier ($p = 0.181$) were not statistically significant, suggesting a performance plateau among top-tier ensemble architectures on this meta-dataset. The MLP classifier also showed a non-significant difference, as it tied for the top rank. Table 8 presents the average rankings of all 18 algorithms across the full benchmark. The bagging classifier and the standard MLP achieved the best average rank of 6.083, closely followed by random forest (6.100).

4.3. Need for automated algorithm selection

Although bagging leads on average, the no free lunch theorem [32] implies that no single classifier dominates every dataset. Under F1-score, 13 of 18 algorithms were top performers on at least one dataset (Figure 3). Simpler models occasionally outperform complex ensembles on specific data structures, underscoring the limitation of a static default choice across heterogeneous healthcare tasks. This study compared a "static best" strategy (always selecting bagging) against an "oracle" strategy (perfectly selecting the

best algorithm per dataset). An automated selector could close a meta-learning gap of 0.0313 average F1-score (approximately 3.1%), a margin that can matter for diagnostic reliability. Together with the performance plateau among top ensembles in subsection 4.2, these results motivate the offline AAS framework developed in this study.

Table 7. Wilcoxon signed-rank tests vs best algorithm (Bagging) for f1_score_mean, with Holm correction

Algorithm	p_value	p_holm	median_diff	mean_diff	win_rate
Ridge classifier	2.113e-05	0.0003592	0.03922	0.1163	0.8667
Nearest centroid	0.0008351	0.01336	0.05696	0.1042	0.7667
Gaussian naive Bayes	0.001322	0.01983	0.06672	0.1135	0.7667
Passive-aggressive classifier	0.001721	0.02409	0.04366	0.0685	0.6333
SGD classifier	0.002672	0.03468	0.03856	0.06853	0.6667
Perceptron	0.00289	0.03468	0.06169	0.07573	0.6667
Linear SVC	0.003938	0.04161	0.02846	0.05444	0.6667
Decision tree	0.004161	0.04161	0.03135	0.04653	0.6333
AdaBoost	0.009726	0.08548	0.009392	0.0632	0.6
Quadratic discriminant analysis	0.01392	0.08548	0.03017	0.04946	0.6667
Logistic regression	0.014	0.08548	0.01193	0.04852	0.5667
K-nearest neighbors	0.01425	0.08548	0.006605	0.03704	0.6333
Linear discriminant Analysis	0.02145	0.1072	0.03108	0.04549	0.6333
NAS-style MLP	0.0409	0.1636	0.008925	0.02194	0.6
Multi-layer perceptron	0.15	0.3616	0.002564	0.01283	0.5
Gradient boosting	0.1808	0.3616	0	0.007178	0.4333
Random forest	0.5872	0.5872	0.0008572	0.004206	0.5

Table 8. Average ranks for f1_score_mean (lower is better)

Algorithm	avg_rank
Bagging	6.083
Multi-layer perceptron	6.083
Random forest	6.100
Gradient boosting	7.133
NAS-style MLP	7.533
Linear discriminant analysis	8.050
AdaBoost	8.967
Logistic regression	9.100
K-nearest neighbors	9.250
Quadratic discriminant analysis	9.467
Linear SVC	9.550
Decision tree	10.000
Perceptron	11.233
SGD classifier	11.350
Passive-aggressive classifier	11.433
Gaussian naive Bayes	13.050
Nearest centroid	13.217
Ridge classifier	13.400

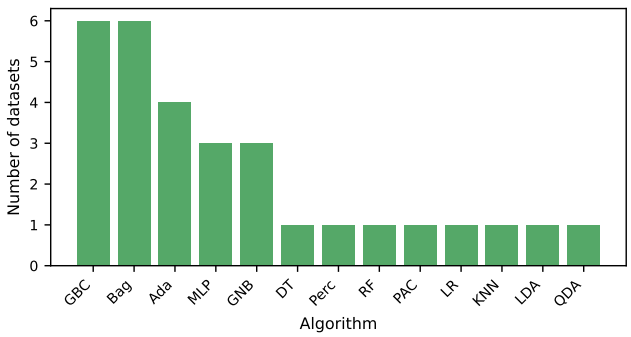

Figure 3. Win count by algorithm based on F1-score

4.4. Class-imbalance sensitivity audit

Healthcare datasets are frequently characterized by significant class imbalance, which can lead to misleadingly high accuracy scores if not properly audited. This study compared the gap between standard accuracy and balanced accuracy across all algorithms and datasets (see Figure 4 and Table 9). A large gap indicates that accuracy looks strong while the model fails on the minority class.

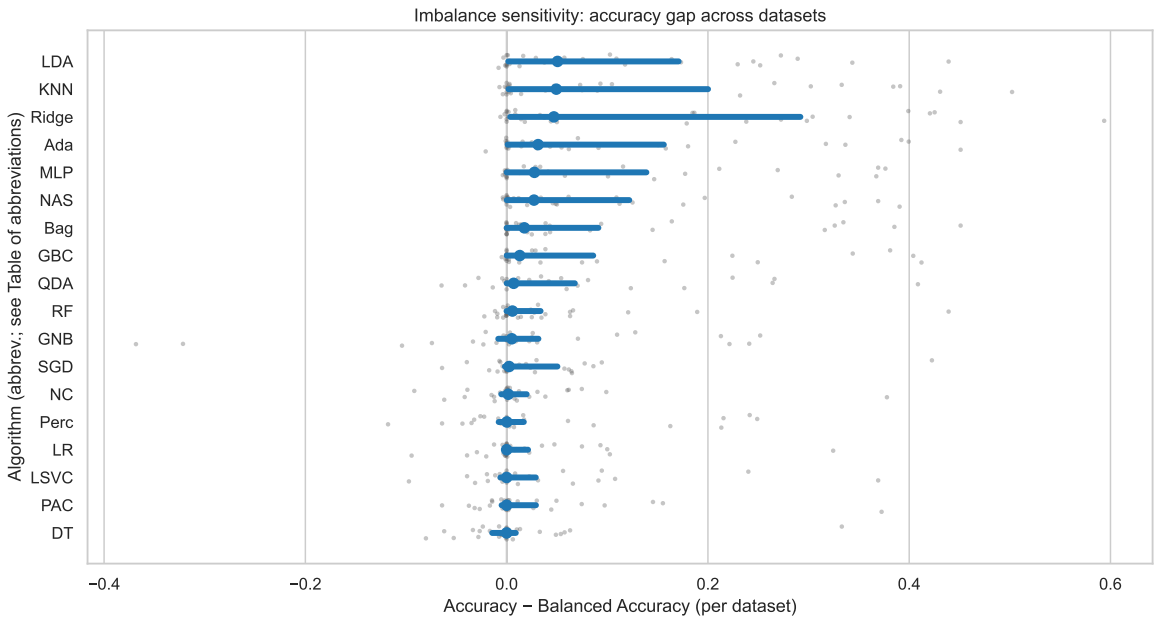

Figure 4. Gap between accuracy and balanced accuracy across datasets for each algorithm

Table 9. Mean, median, and standard deviation of the accuracy gap for each classification algorithm

Algorithm	mean	median	std
Linear discriminant analysis	0.102	0.051	0.124
K-nearest neighbors	0.116	0.049	0.156
Ridge classifier	0.154	0.047	0.177
AdaBoost	0.101	0.031	0.141
Multi-layer perceptron	0.090	0.028	0.129
NAS-style MLP	0.090	0.027	0.127
Bagging	0.085	0.018	0.135
Gradient boosting	0.085	0.013	0.136
Quadratic discriminant analysis	0.057	0.007	0.107
Random forest	0.037	0.006	0.088
Gaussian naive Bayes	0.015	0.005	0.132
SGD classifier	0.030	0.002	0.083
Nearest centroid	0.017	0.002	0.078
Perceptron	0.031	0.000	0.092
Logistic regression	0.021	0.000	0.070
Linear SVC	0.029	0.000	0.087
Passive-aggressive classifier	0.027	0.000	0.081
Decision tree	0.009	0.000	0.069

The analysis shows on one hand that that simpler models and linear classifiers show a wider gap, which suggests a higher sensitivity to imbalance. For example, ridge classifier and k-nearest neighbors displayed median gaps of 0.047 and 0.049 respectively, with much higher mean deviations exceeding 0.11. Ridge also shows lower macro sensitivity ($\approx$ 0.72, Table 4). On the other hand, ensemble methods like the bagging classifier and random forest maintain significantly narrower median gaps (0.018 and 0.006 respectively),

further justifying their selection as preferred models for clinical applications where minority class detection (e.g., presence of a rare disease) is paramount.

4.5. Interpretability: linking meta-features to algorithm performance

The choice of algorithms rests on the relative performances resulting as theorized in (5) and (6). In order to understand which features affect the predictions most, the correlation between the meta-features and F1-score measures were analyzed, and the results are summarized in Table 10 and Figure 5. The selected meta-features include mean attribute entropy, number of instances, number of attributes, class frequency variation, mean sparsity, and class entropy. These correlations show which dataset properties most influence predicted F1-score, and therefore which descriptors later help explain a recommendation.

Table 10. Spearman ρ between selected meta-features and algorithm F1-scores

Algorithm	Attr. Ent.	Inst.	Attr.	Class Freq.	Sparsity mean	Class entropy
AdaBoost	0.339	-0.106	0.353	-0.081	-0.427	0.015
Bagging	0.170	-0.184	0.160	-0.413	-0.422	0.403
Decision tree	0.251	-0.182	0.142	-0.280	-0.394	0.257
Gradient boosting	0.147	-0.120	0.153	-0.346	-0.374	0.343
Gaussian naive Bayes	0.250	-0.236	0.014	-0.549	-0.504	0.436
K-nearest neighbors	0.249	-0.241	0.029	-0.589	-0.530	0.576
Linear discriminant analysis	0.120	-0.373	0.002	-0.540	-0.417	0.469
Logistic regression	0.079	-0.396	-0.049	-0.596	-0.383	0.521
Linear SVC	0.146	-0.325	0.028	-0.594	-0.418	0.524
Multi-layer perceptron	0.217	-0.199	0.123	-0.494	-0.474	0.485
NAS-style MLP	0.221	-0.182	0.106	-0.507	-0.481	0.492
Nearest centroid	-0.057	-0.448	-0.130	-0.586	-0.244	0.486
Passive-aggressive classifier	0.121	-0.377	0.015	-0.533	-0.376	0.475
Perceptron	0.151	-0.332	0.072	-0.502	-0.358	0.411
Quadratic discriminant analysis	0.182	-0.278	0.052	-0.563	-0.478	0.494
Random forest	0.201	-0.223	0.098	-0.465	-0.446	0.459
Ridge classifier	0.223	-0.419	0.007	-0.542	-0.438	0.468
SGD classifier	0.126	-0.255	0.105	-0.532	-0.365	0.443

The analysis shows many notable patterns. Ensemble methods such as random forest and bagging classifier present a positive correlation with class entropy (rho $\approx$ 0.40–0.46) and a negative correlation with data sparsity (rho ≈ -0.42–-0.45). This suggests that these algorithms perform better on datasets with well distributed classes and struggle as data becomes sparse. Linear models (logistic regression, linear SVC) show even stronger sensitivity to to class frequency standard deviation (rho ≈ -0.60). These findings align with the audit we conducted in subsection 4.4. The recommender's top ranked choice shifts accordingly with these results.

This study also identified the 10 meta-features presenting the strongest correlations with the performance of bagging classifier as can see in Table 11. Certain landmarks and information-theoretic measures such as Roy's root (rho = 0.72) and the canonical correlation (rho = 0.71), show strong association with F1-score. Inversely, nodes per instance presents a strong negative correlation (rho = -0.77) suggesting that the performance of ensemble methods tends to degrade the higher the complexity required of a simple decision tree to solve a problem is.

These insights move beyond a 'black-box' selection process, summarizing which meta-features most influence algorithm choice. For example, high tree-complexity landmarks such as nodes per instance or low Roy's root, predict lower bagging F1-score and often leave another model atop the recommended rank. If a dataset is flagged with high sparsity and imbalance, the system can transparently favor robust ensemble methods while warning against simpler linear baselines [13].

4.6. Regression and ranking evaluation

This study trained regression models (Ridge, Lasso, ElasticNet, random forest, and k-nearest neighbors) for each classifier and target metric under the LODO protocol in section 3. For every metric, we fit 18 dataset-level meta-models (one per algorithm), select the regressor family with highest leave-one-out (LOO) R^2, and aggregate errors across algorithms. Table 12 reports mean and standard deviation of LOO R^2, MAE, and MSE across those 18 fits: accuracy is the most predictable target (mean $R^2 = 0.61 \pm 0.21$), whereas F1-score, precision, and recall show lower and more variable fit ($R^2 \approx 0.38$–0.45), consistent with

class-imbalance sensitivity and the small meta-sample ($N = 30$). This study treat dispersion in Table 12 as the primary robustness indicator for meta-regression; with only 30 meta-instances, bootstrap confidence intervals on a pooled R^2 would be unstable and are not reported. Ranking quality (Table 13) is evaluated on the same LOO predictions.

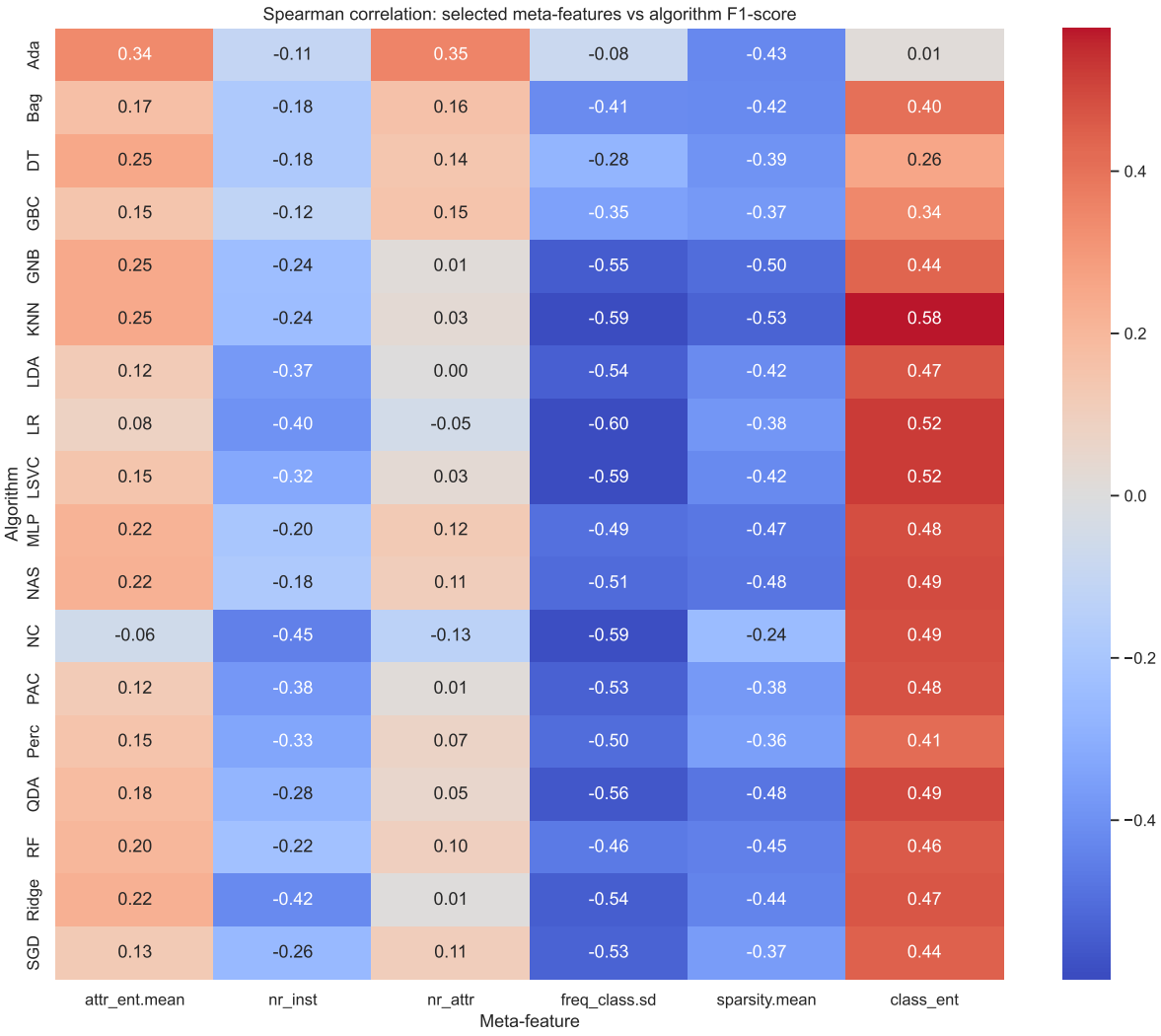

Figure 5. Spearman correlations between selected meta-features and the F1-score of each classification algorithm

Table 11. Top meta-features most correlated with Bagging F1-score

Meta-feature	Spearman rho	P-value	N
Nodes per instance	-0.773	0.000	30
Relative worst-node performance	-0.750	0.000	30
Fisher ratio variability	0.741	0.000	26
Mean repeated nodes	-0.738	0.000	30
Nodes per attribute	-0.733	0.000	30
Roy's largest root	0.724	0.000	30
Lawley–hotelling trace	0.716	0.000	30
Mean feature efficiency	-0.714	0.000	30
Mean nodes per tree level	-0.711	0.000	30
Mean canonical correlation	0.709	0.000	30

Table 12. Overall summary of model performance by metric

Metric	Avg R2	Std R2	Avg MAE	Std MAE	Avg MSE	Std MSE
Accuracy	0.613	0.209	0.048	0.021	0.006	0.006
F1 score	0.452	0.262	0.091	0.032	0.016	0.010
Precision	0.378	0.261	0.086	0.026	0.014	0.009
Recall	0.389	0.198	0.087	0.026	0.014	0.008

For algorithm recommendation, this study use learning-to-rank (LTR) with LambdaRank to optimize list-wise ordering from predicted scores. Table 13 reports NDCG@k against random selection and a global-best heuristic. The LTR recommender achieves NDCG@1 above 0.97 for accuracy and above 0.94 for F1-score, selects the true top-1 algorithm in 66.67% of F1 cases (83.33% within top-2), and outperforms random selection at all cutoffs; it remains competitive with the global-best heuristic as ranking depth increases, especially for F1 under dataset-specific imbalance. Statistical significance of classifier performance differences is assessed separately in subsection 4.2 (Friedman and Holm-corrected Wilcoxon tests on observed benchmark scores).

Table 13. NDCG ranking quality vs. random and global-best baselines

Metric	Strategy	NDCG@1	NDCG@3	NDCG@5
Accuracy	AAS (LTR)	0.9750	0.9805	0.9829
Accuracy	Random	0.9098	0.9188	0.9263
Accuracy	Global-best heuristic	0.9855	0.9778	0.9562
F1-score	AAS (LTR)	0.9491	0.9506	0.9521
F1-score	Random	0.8812	0.8911	0.9071
F1-score	Global-best heuristic	0.9399	0.9331	0.9267

4.7. Practical implications and discussion

Let's take the example of a practitioner whose goal is to analyse a diabetes dataset but who lacks the necessary skills to choose from the multitude of available classifiers which one would suit his specific problem. Traditionally, this would either require a lot of trial and error or the intervention of an expert. With our system, the practitioner uploads the dataset, meta-features get automatically extracted, then sent to the pretrained regression model. The system then produces a list of rakings for the available algorithms from which the user can choose from. This simplifies the process and democratizes advanced analytics tools for clinics with limited technical support.

In a hospital environment, the user base can be clinical analytics experts, biostatisticians or IT teams rather than an autonomous decision system. For such environments, the proposed deployment process is as follows: First, export a anonymized dataset from the electronic health record (EHR) or a clinical database. Second, apply the unified pretraining formula as explained in section 3. Third, extract meta-features using available libraries (e.g. PyMFE). Fourth, use the meta regressor to produce a ranking of the selected algorithms based on the meta-features. Fifth, validate the top algorithms by running a cross-validated test using the exported dataset for final algorithm approval and deployment. However, a direct integration with EHR flows would require adaptation on the level of the establishment, governance review and protected health information (PHI) safeguards. The evaluated protocol in this work is based on already prepared tabular datasets with a clear classification problem each time, an interfacing with live data would require extra preparation and adaptation. It is also important to note that the system is a tool of support for decision making: it suggests candidate classifiers and provides explanation elements based on the features of a dataset, while clinicians retain responsibility for the final validation and deployment.

The NAS-style MLP yielded decent results compared to the ensemble methods but was not as consistent as bagging on the tests we ran. These results suggest that as expected, robust ensemble methods remain the reference option for mid-size datasets. While other approaches used learned patterns or visual representations [22], [33], [34], the method used tabular meta-features, making it easier to interpret results and better adapted to the limited dimensions of clinical datasets. Correlation analysis in subsection 4.5. supports transparency of recommendations in healthcare settings where explainability is key.

This study still presents some limitations: used a pool of 30 publicly available datasets (Table 1), some of which did not provide complete geographical or demographics information as detailed in subsection 3.1. Results need to be further validated using independent clinical datasets, in the context of a pilot evaluation of EHR deployment workflows. Nested cross validation and LODO might help limit the risks of data leakage but the smaller size of the meta dataset increases the variability of the predictions, a bigger selection of used datasets

would solve this problem (synthetic data can be considered if carefully constructed to represent clinical data). Finally, although we used MLPs as a first step to represent deep learning techniques, more advanced methods such as DeepGBMs, DANs or transformers were not explored for computational cost reasons.

5. CONCLUSIONS

This work presents an approach based on meta-learning for automatic algorithm selection applied to healthcare datasets. It uses 18 algorithms, 30 public datasets and a set of extracted meta-features for each dataset. The proposed system allows to predict the top performing algorithms based using F1-score as the selection criterion. Future works will focus on reinforcing the robustness and applicability of the system by expanding the dataset pool, including other subfields such as genomics, proteomics or medical imaging where the use of synthetic data generation is becoming standard practice for training models. They will also explore adding more deep learning methods in the zoo of predictors but also as meta regressors, incorporating uncertainy quantification like conformal predictions or Bayesian model averaging to associate confidence scores to the recommendations and conducting pilot experiments in practical settings to evaluate impact on the workflows and real diagnostic utility.

FUNDING INFORMATION

The authors state no funding involved.

AUTHOR CONTRIBUTIONS

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Mohamed Abdelbaki	✓	✓	✓		✓	✓		✓	✓		✓			
Adnane Latif	✓			✓			✓			✓		✓	✓	
Charafeddine Ait Zaouiat		✓		✓						✓				

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal Analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject Administration
Fu : **Fu**nding Acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

The datasets used in this study are publicly available from the primary sources listed in Table 1. The curated benchmark registry and experiment metadata are available from the corresponding author upon request.

REFERENCES

[1] J. R. Rice, "The algorithm selection problem," in *Advances in Computers*, vol. 15, Elsevier, 1976, pp. 65–118, doi: 10.1016/S0065-2458(08)60520-3.
[2] K. A. S.-Miles, "Cross-disciplinary perspectives on meta-learning for algorithm selection," *ACM Computing Surveys*, vol. 41, no. 1, pp. 1–25, Jan. 2009, doi: 10.1145/1456650.1456656.
[3] P. Kerschke, H. H. Hoos, F. Neumann, and H. Trautmann, "Automated algorithm selection: survey and perspectives," *Evolutionary Computation*, vol. 27, no. 1, pp. 3–45, Mar. 2019, doi: 10.1162/evco_a_00242.
[4] X. He, K. Zhao, and X. Chu, "AutoML: a survey of the state-of-the-art," *Knowledge-Based Systems*, vol. 212, Jan. 2021, doi: 10.1016/j.knosys.2020.106622.
[5] 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, Apr. 2020, doi: 10.1016/j.artmed.2020.101822.

[6] A. Mustafa and M. R. Azghadi, "Automated machine learning for healthcare and clinical notes analysis," *Computers*, vol. 10, no. 2, Feb. 2021, doi: 10.3390/computers10020024.
[7] I. H. Sarker, "Machine learning: algorithms, real-world applications and research directions," *SN Computer Science*, vol. 2, no. 3, Mar. 2021, doi: 10.1007/s42979-021-00592-x.
[8] F. Imrie, B. Cebere, E. F. McKinney, and M. V. D. Schaar, "AutoPrognosis 2.0: democratizing diagnostic and prognostic modeling in healthcare with automated machine learning," *PLOS Digital Health*, vol. 2, no. 6, 2023, doi: 10.1371/journal.pdig.0000276.
[9] Y. Bai, Y. Li, Y. Shen, M. Yang, W. Zhang, and B. Cui, "AutoDC: An automatic machine learning framework for disease classification," *Bioinformatics*, vol. 38, no. 13, pp. 3415–3421, 2022, doi: 10.1093/bioinformatics/btac334.
[10] L. M. Paladino, A. Hughes, A. Perera, O. Topsakal, and T. C. Akinci, "Evaluating the performance of automated machine learning (AutoML) tools for heart disease diagnosis and prediction," *AI*, vol. 4, no. 4, pp. 1018–1034, 2023, doi: 10.3390/ai4040053.
[11] G. Cenikj, A. Nikolikj, G. Petelin, N. V. Stein, C. Doerr, and T. Eftimov, "A survey of meta-features used for automated selection of algorithms for black-box single-objective continuous optimization," 2024, *arXiv:2406.06629*.
[12] I. Khan, X. Zhang, M. Rehman, and R. Ali, "A literature survey and empirical study of meta-learning for classifier selection," *IEEE Access*, vol. 8, pp. 10262–10281, 2020, doi: 10.1109/ACCESS.2020.2964726.
[13] X. Shao, H. Wang, X. Zhu, F. Xiong, T. Mu, and Y. Zhang, "EFFECT: Explainable framework for meta-learning in automatic classification algorithm selection," *Information Sciences*, vol. 622, pp. 211–234, 2023, doi: 10.1016/j.ins.2022.11.144.
[14] E. Alcobaça, F. Siqueira, A. Rivolli, L. P. F. Garcia, A. C. P. L. F. D. Carvalho, and J. Vanschoren, "MFE: towards reproducible meta-feature extraction," *Journal of Machine Learning Research*, vol. 21, no. 111, pp. 1–5, Jan. 2020, doi: 10.5555/3455716.3455827.
[15] M. Garouani, A. Ahmad, M. Bouneffa, M. Hamlich, G. Bourguin, and A. Lewandowski, "Using meta-learning for automated algorithms selection and configuration: an experimental framework for industrial big data," *Journal of Big Data*, vol. 9, no. 1, Apr. 2022, doi: 10.1186/s40537-022-00612-4.
[16] M. Garouani, A. Ahmad, M. Bouneffa, and M. Hamlich, "Autoencoder-kNN meta-model based data characterization approach for an automated selection of AI algorithms," *Journal of Big Data*, vol. 10, 2023, doi: 10.1186/s40537-023-00687-7.
[17] L. Li, Y. Wang, Y. Xu, and K.-Y. Lin, "Meta-learning based industrial intelligence of feature nearest algorithm selection framework for classification problems," *Journal of Manufacturing Systems*, vol. 62, pp. 767–776, 2022, doi: 10.1016/j.jmsy.2021.03.007.
[18] P. Golshanrad, H. Rahmani, B. Karimian, F. Karimkhani, and G. Weiss, "MEGA: predicting the best classifier combination using meta-learning and a genetic algorithm," *Intelligent Data Analysis*, vol. 25, no. 6, pp. 1547–1563, Jan. 2021, doi: 10.3233/IDA-205494.
[19] I. Khan, X. Zhang, R. K. Ayyasamy, and R. Ali, "AutoFE-Sel: a meta-learning based methodology for recommending feature subset selection algorithms," *KSII Transactions on Internet and Information Systems*, vol. 17, no. 7, pp. 1773–1793, 2023, doi: 10.3837/tiis.2023.07.002.
[20] Ş. Ay, E. Ekinci, and Z. Garip, "A comparative analysis of meta-heuristic optimization algorithms for feature selection on ML-based classification of heart-related diseases," *The Journal of Supercomputing*, vol. 79, no. 11, pp. 11797–11826, Mar. 2023, doi: 10.1007/s11227-023-05132-3.
[21] A. Bogatu *et al*., "Meta-analysis informed machine learning: supporting cytokine storm detection during CAR-T cell therapy," *Journal of Biomedical Informatics*, vol. 142, 2023, doi: 10.1016/j.jbi.2023.104367.
[22] I. Dagan, R. Vainshtein, G. Katz, and L. Rokach, "Automated algorithm selection using meta-learning and pre-trained deep convolution neural networks," *Information Fusion*, vol. 105, May 2024, doi: 10.1016/j.inffus.2023.102210.
[23] M. Zhabinets, B. Tyler, M. Lukac, S. Nagayama, F. Molnár, and M. Kameyama, "Synthetic data-based algorithm selection for medical image classification under limited data availability," *Algorithms*, vol. 18, no. 6, 2025, doi: 10.3390/a18060310.
[24] M. Feurer, A. Klein, K. Eggensperger, J. T. Springenberg, M. Blum, and F. Hutter, "Efficient and robust automated machine learning," in *NIPS'15: Proceedings of the 28th International Conference on Neural Information Processing Systems*, 2015, pp. 2755–2763, doi: 10.5555/2969442.2969547.
[25] W. J. Clancey, "Classification problem solving," *Artificial Intelligence*, vol. 24, no. 1–3, pp. 289–350, 1984, doi: 10.1016/0004-3702(84)90044-5.
[26] C. Ferri, J. Hernández-Orallo, and R. Modroiu, "An experimental comparison of performance measures for classification," *Pattern Recognition Letters*, vol. 30, no. 1, pp. 27–38, Jan. 2009, doi: 10.1016/j.patrec.2008.08.010.
[27] D. J. Cook and R. C. Varnell, "Maximizing the benefits of parallel search using machine learning," in *Proceedings of the fourteenth national conference on artificial intelligence and ninth conference on Innovative applications of artificial intelligence*, 1997, pp. 559–564.
[28] D. Michie, D. J. Spiegelhalter, and C. C. Taylor, *Machine learning, neural and statistical classification*. Chichester, United Kingdom: Ellis Horwood, 1994.
[29] L. Xu, F. Hutter, J. Shen, H. Hoos, and K. L. -Brown, "SATzilla2012: improved algorithm selection based on cost-sensitive classification models," in *Proceedings of SAT Challenge 2012: Solver and Benchmark Descriptions*, 2012, pp. 55–58.
[30] M. Lindauer, H. H. Hoos, F. Hutter, and T. Schaub, "AutoFolio: an automatically configured algorithm selector," *Journal of Artificial Intelligence Research*, vol. 53, pp. 745–778, Aug. 2015, doi: 10.1613/jair.4726.
[31] M. Lindauer, F. Hutter, H. H. Hoos, and T. Schaub, "AutoFolio: an automatically configured algorithm selector (extended abstract)," in *Twenty-Sixth International Joint Conference on Artificial Intelligence*, Aug. 2017, pp. 5025–5029, doi: 10.24963/ijcai.2017/715.
[32] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, Apr. 1997, doi: 10.1109/4235.585893.
[33] M. Seiler, J. Pohl, J. Bossek, P. Kerschke, and H. Trautmann, "Deep learning as a competitive feature-free approach for automated algorithm selection on the traveling salesperson problem," in *International Conference on Parallel Problem Solving from Nature*, 2020, pp. 48–64, doi: 10.1007/978-3-030-58112-1_4.
[34] S. Maldonado, C. Vairetti, and I. Figueroa, "Automatic learning algorithm selection for classification via convolutional neural networks," May 2023, *arXiv.2305.09101*.

BIOGRAPHIES OF AUTHORS



Mohamed Abdelbaki is a computer science engineer and researcher specializing in software engineering with a focus on the application of artificial intelligence in healthcare. He obtained his engineering diploma from the National School of Applied Sciences of Marrakech and developed a keen interest in utilizing advanced technologies to enhance healthcare systems. During his studies, he embarked on a significant project developing a web application to visualize machine learning models used in healthcare, aiming to facilitate informed decision-making and advance AI in the medical field. He served as an instructor and as an assistant for various subjects including web development, microcontrollers, and computer architecture. He is currently pursuing a Ph.D. at the Laboratory of Complex Systems Modeling, National School of Applied Sciences of Marrakech, Cadi Ayyad University, Morocco. His research revolves around automating the selection of machine learning algorithms for healthcare applications using meta-learning techniques. His ultimate goal is to develop intelligent agents that aid in making informed decisions, leveraging the latest algorithms and techniques inspired by advancements in large language models. By harnessing the power of artificial intelligence, he envisions a future where healthcare systems become more efficient, accurate, and accessible, benefiting both patients and healthcare providers. He can be contacted at email: abdelbaki.mo6@gmail.com.



Adnane Latif has a Ph.D. in Telecommunications and Networks, professor at Laboratory of Complex Systems Modeling, National School of Applied Sciences, Cadi Ayyad University, Marrakesh, Morocco. Researcher in the areas of antennas, microwaves and wireless communications. Author of a fifties publications in international conferences and in international journals. He is founder and coordinator of the spring school 2010 “Wireless and Emerging Technologies”. He is founder and president of the Moroccan Association of Telecommunications and Electronics Technologies. He is member organizer of several international conferences and spring schools. He is member of the international program committee of several international conferences. He is reviewer and board member of several international journals. He is author of a book entitled “Patch antenna design and optimization: applications in wireless networks”. He can be contacted at email: a.latif@uca.ac.ma.



Charafeddine Ait Zaouiat is specializes in networks and telecommunications, with a particular interest in the application of the internet of things (IoT) and artificial intelligence in healthcare. In 2010, he obtained his engineering degree in Networks and Telecommunications, which provided him with a technical background in communication systems, network infrastructures, and connected technologies. He later continued his academic studies in the same field and completed a Ph.D. focused on the use of IoT and AI techniques in healthcare. His doctoral research examined how connected devices and intelligent systems can be used to collect, transmit, process, and analyze health data, with the objective of improving healthcare services and supporting better patient outcomes. He is currently a professor of Information Technology at Chouaib Doukkali University, where he contributes to teaching, academic supervision, and research activities. His research mainly covers IoT, artificial intelligence, communication technologies, and their applications in the healthcare domain. More specifically, his work explores the use of these technologies for medical diagnosis support, remote patient monitoring, connected medical environments, personalized care, and the secure exchange of healthcare information. Throughout his academic career, he has contributed to several research projects and publications in collaboration with other researchers in the field. His work follows a practical objective: developing technological solutions that can respond to real healthcare challenges and improve the way medical data is collected, transmitted, analyzed, and used in clinical and remote-care environments. He can be contacted at email: charafeddineaitzaouiat@gmail.com.