Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A multi-objective evolutionary approach to neural architecture search for clinical tabular classification: balancing predictive performance and model compactness

Ivan V. Stepanyan
Mechanical Engineering Research Institute of the Russian Academy of Sciences (IMASH RAN)
Menhai Hou
houmenhai@gmail.com
Peoples' Friendship University of Russia named after Patrice Lumumba (RUDN University)
Safa A. Hameed
Peoples' Friendship University of Russia named after Patrice Lumumba (RUDN University)

Article

Keywords: Neural architecture search, Multi-objective optimization, Genetic algorithm, NSGA-II, Pareto front, Model compactness

Posted Date: July 14th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10092066/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Abstract

Designing compact neural network classifiers for clinical tabular data is difficult: manual tuning is labour-intensive, and single-objective optimization tends to favour unnecessarily large models that generalize poorly and are awkward to deploy on constrained hardware. This paper presents a multi-objective genetic algorithm for neural architecture search (MOGA-NAS) that simultaneously maximizes the F1-score and minimizes the parameter count using NSGA-II-based non-dominated sorting. The method incorporates a neuron-to-subnetwork mutation operator, an importance-guided neuron-selection scheme, and a compact 8-bit encoding of layer widths. To obtain unbiased estimates and to avoid information leakage from the test partition into the search process, all architectures are evaluated under a nested cross-validation protocol with an inner validation split driving the evolutionary fitness and an outer fold reserved strictly for testing; each configuration is repeated over three random seeds (fifteen runs in total). On five public clinical benchmarks (Cleveland Heart Disease, Wisconsin Breast Cancer, Indian Liver Patient, Diabetic Retinopathy Debrecen, and Heart Failure Clinical Records), MOGA-NAS attains the highest mean F1-score on four datasets and is statistically indistinguishable from the strongest baseline on the fifth, while producing models of the order of one hundred parameters—one to two orders of magnitude smaller than several competing variants. Paired Wilcoxon signed-rank tests confirm that the improvements are statistically significant in the large majority of comparisons, and an ablation study indicates that each algorithmic component contributes positively. The results suggest that multi-objective evolutionary search offers a favourable and reproducible trade-off between predictive performance and model compactness for tabular clinical classification.

1. Introduction

Machine learning has become a routine tool in clinical decision support, and neural networks in particular have shown promising results across many binary classification tasks such as diabetes prediction, heart-disease detection, and tumour classification [1, 2]. Their performance, however, is highly sensitive to architectural choices: the number of layers, the number of units per layer, and the connectivity pattern all materially affect the outcome [3]. Designing these architectures by hand demands expertise and repeated trial and error, which is inefficient and difficult to reproduce.

Neural architecture search (NAS) automates this design process [4, 5]. Among the available strategies, evolutionary algorithms are attractive because their population-based search explores diverse configurations without requiring gradient information [6, 7]. Genetic algorithms (GAs), a widely used subclass, evolve network topologies through selection, crossover, and mutation [8, 9]. Most GA-based NAS methods, however, optimize a single objective—typically accuracy or loss [10]. This tends to drive the search towards increasingly large architectures that may overfit and that are awkward to deploy on portable or embedded clinical hardware [11, 12]. In practice, a clinical model must balance two potentially conflicting goals: strong predictive performance and low computational cost.

Multi-objective optimization addresses this tension directly by treating the conflicting goals jointly [13]. The Non-dominated Sorting Genetic Algorithm II (NSGA-II) [14] is a well-established multi-objective evolutionary algorithm that uses Pareto-based ranking and crowding distance to maintain a diverse set of trade-off solutions. NSGA-II has been applied to NAS for computer-vision tasks [15, 16], but its use for tabular clinical data with evolved fully-connected architectures remains comparatively underexplored.

This paper studies a multi-objective genetic algorithm for neural architecture search, denoted MOGA-NAS, that combines NSGA-II selection with three design elements: a neuron-to-subnetwork mutation operator that lets individual neurons expand into small multi-layer sub-modules, an importance-guided scheme for selecting which neurons to mutate, and a compact 8-bit encoding of layer widths. We do not claim a new state of the art; rather, we aim to characterize, under a careful and reproducible evaluation protocol, the trade-off that such a method offers between F1-score and model size on clinical tabular benchmarks. The contributions are as follows.

1. **Multi-objective formulation.** F1-score and parameter count are optimized jointly with NSGA-II, yielding a Pareto front of architectures that expose different performance–complexity trade-offs and allow a model to be chosen according to deployment constraints.
2. **Architectural and encoding operators.** A neuron-to-subnetwork mutation operator and an importance-guided neuron-selection scheme are combined with a compact binary encoding; each is examined individually through an ablation study.
3. **Leakage-free, reproducible evaluation.** All architectures are assessed under a nested cross-validation protocol that isolates the test fold from the search, repeated over three seeds, with statistical testing (paired Wilcoxon signed-rank), bootstrap confidence intervals, and standard Pareto-quality indicators (hypervolume, inverted generational distance, and spread).

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 describes the methodology. Section 4 details the experimental setup. Section 5 reports and discusses the results, including ablation and Pareto-quality analyses. Section 6 states the limitations, and Section 7 concludes.

2. Related Work

2.1 Evolutionary Neural Architecture Search

Evolutionary methods have a long history in neural network design. Yao [17, 18] showed that genetic algorithms can evolve both the topology and the weights of neural networks. With the resurgence of deep learning, evolutionary NAS regained attention: Real et al. [19] applied large-scale evolution to image classifiers, and later introduced an aging mechanism to accelerate the search [16]; Sun et al. [10, 20] proposed a flexible genetic encoding for automatically designing convolutional architectures. Stanley et al. [8] reviewed neuroevolution and highlighted its ability to discover non-intuitive structures. In clinical applications, Aquino et al. [21] used neuro-evolutionary algorithms for diabetic-retinopathy diagnosis from fundus images, Wijaya et al. [23] compared NEAT with deep belief networks for diabetes detection,

and Kumar et al. [22] combined evolutionary computation with ensemble models. These studies, however, almost universally adopt a single optimization objective and do not explicitly control model complexity, so the search may yield architectures that are larger than necessary for resource-constrained deployment.

2.2 Multi-Objective Neural Architecture Search

Multi-objective NAS recognizes that deployment requires balancing several criteria. Lu et al. [15] proposed NSGA-Net, combining NSGA-II with a Bayesian-network-assisted exploitation step to optimize accuracy and computational cost (FLOPs) jointly for image classification. Elsken et al. [24] introduced a multi-objective evolutionary method that trades predictive performance against architectural complexity, using Lamarckian inheritance to speed up convergence. Subsequent work has explored surrogate models and training-free proxies to reduce search cost, and recent surveys [5, 25] identify multi-objective evolutionary NAS as a growing direction, especially for resource-limited settings. Most of this literature, however, targets convolutional architectures and image data; multi-objective architecture search for fully-connected networks on tabular clinical data remains comparatively under-studied.

2.3 Genetic Optimization in Medical Diagnosis

Genetic algorithms have been used in several medical-classification studies. Chowdary et al. [26] combined a neuro-fuzzy feature-extraction model with genetic optimization for diabetes classification. Al-Badarneh et al. [27] trained multilayer perceptrons with three metaheuristics (Grey Wolf Optimization, Particle Swarm Optimization, and the Salp Swarm Algorithm) on ten imbalanced benchmarks, and reported that optimizing the F1-score or G-mean as the fitness function is preferable to optimizing accuracy under class imbalance—a finding that motivates the choice of F1-score as the primary objective here. Buccheri et al. [12] emphasized the value of model simplicity for large-scale screening, and Galvan and Mooney [7] surveyed neuroevolution and noted search efficiency, scalability, and multi-objective optimization as open challenges.

Against this background, the present work is positioned as follows: relative to single-objective evolutionary NAS, it introduces NSGA-II selection to control model size while optimizing performance; relative to existing multi-objective NAS, it focuses on tabular clinical data with fully-connected architectures; and relative to conventional GA hyperparameter tuning, the neuron-to-subnetwork operator provides additional architectural flexibility. The study is deliberately framed as a careful empirical characterization rather than a claim of broad superiority.

3. Methodology

3.1 Overview and Problem Formulation

MOGA-NAS maintains a population of candidate fully-connected architectures that undergo evolutionary operations. In each generation, individuals pass through five stages: mutation, training, evaluation,

selection, and crossover. Selection uses NSGA-II and considers two objectives simultaneously: maximizing the validation F1-score and minimizing the number of trainable parameters.

Let a denote a candidate architecture drawn from the search space A. The two competing objectives are the validation F1-score f1(a) and the parameter count P(a). The search is formulated as the bi-objective optimization problem

$$\max a \in A \; F(a) = (f_1(a), -P(a)), \quad (1)$$

where the minus sign expresses parameter minimization as a maximization so that both components share a common direction. The F1-score is the harmonic mean of precision and recall,

$$f_1 = 2 \cdot (Pr \cdot Re) / (Pr + Re), \; Pr = TP/(TP + FP), \; Re = TP/(TP + FN), \quad (2)$$

with TP, FP, and FN the numbers of true positives, false positives, and false negatives. For a network with L hidden layers and weight matrices Wl and bias vectors bl, the parameter count is

$$P(a) = \Sigma_l (|W_l| + |b_l|), \; l = 1, \ldots, L + 1. \quad (3)$$

Because the two objectives are generally in conflict—larger models can fit the training data more closely but cost more and may generalize worse—there is no single optimum but a set of non-dominated trade-offs, the Pareto front.

3.2 Architecture Encoding

Each individual encodes a variable-depth multilayer perceptron with between one and five hidden layers. Layer widths are represented with a compact 8-bit binary code per layer, bounding each width to the range used in the search (8 to 128 units) and providing a discrete, mutation-friendly representation. Two alternative encodings—an integer encoding and a real-valued encoding—are also implemented and are examined in the ablation study (Section 5.4); as shown there, the binary encoding tends to behave as an implicit regularizer by constraining the effective search space. The genotype additionally carries, for each neuron, an optional flag indicating whether that neuron has been expanded into a subnetwork (Section 3.3).

3.3 Neuron-to-Subnetwork Mutation

The central architectural operator allows an individual neuron to expand into a small multi-layer subnetwork during evolution. When a neuron is selected for expansion, its single linear unit is replaced by a compact two- or three-layer module whose output is projected back to a scalar, so the surrounding architecture remains valid. This mechanism lets the search introduce localized hierarchical capacity only where it is useful, instead of uniformly widening or deepening the whole network. The expansion is reversible through the complementary mutation, so the operator can both add and remove local capacity over generations.

3.4 Importance-Guided Neuron Selection

Rather than selecting neurons for mutation uniformly at random, MOGA-NAS estimates a simple importance score for each neuron and biases mutation towards more influential units. The importance of neuron i is approximated by the change in the training loss L when the neuron's output is temporarily set to zero,

$I(i) = | L(a ; h_i = 0) - L(a) |$, (4)

where hi is the activation of neuron i. This perturbation-based estimate requires only additional forward passes and no extra back-propagation. The probability of selecting neuron i for mutation is then taken proportional to its importance, pi = I(i) / Σj I(j). The ablation study contrasts this scheme with uniform random selection to quantify its contribution.

3.5 NSGA-II Selection

Selection follows the standard NSGA-II procedure [14]. A solution a is said to dominate a solution b, written a ≻ b, if it is no worse in every objective and strictly better in at least one,

$a \succ b \Leftrightarrow \forall k: F_k(a) \geq F_k(b) \wedge \exists k: F_k(a) > F_k(b)$. (5)

The combined parent-and-offspring population is sorted into non-domination fronts. Within a front, diversity is preserved using the crowding distance, which for solution i sums the normalized objective gaps to its nearest neighbours,

$d_i = \Sigma_k (F_k(i+1) - F_k(i-1)) / (F_k^{max} - F_k^{min})$, (6)

where the neighbours are defined by sorting the front along objective k. Individuals are chosen front by front, and, when a front must be split, those with larger crowding distance are preferred. As an ablation, a scalarized single-objective variant replaces the Pareto ranking with fitness based on a weighted sum,

$g(a) = \alpha \cdot f_1(a) - (1 - \alpha) \cdot P(a), \alpha \in [0, 1]$, (7)

with P the normalized parameter count; this variant, together with one that optimizes F1 alone, is examined in Section 5.4.

3.6 Evaluation Protocol and Information-Leakage Control

A central methodological concern in NAS is that the data used to guide the search must be kept separate from the data used to report performance; otherwise the reported scores are optimistically biased. To prevent such leakage, architectures are assessed under a nested cross-validation protocol. The outer loop is a stratified five-fold partition; in each outer fold, the test partition is held out and never used during evolution. Within the remaining data, an inner training/validation split provides the validation F1-score that drives the evolutionary fitness. After the search concludes, the selected architecture is retrained on the combined inner training and validation data and evaluated once on the untouched outer

test fold. Feature standardization is fitted only on the inner training partition and then applied to the validation and test partitions, so no test-set statistics enter the pipeline.

To account for stochasticity in both the evolutionary search and network initialization, the entire protocol is repeated with three random seeds, giving fifteen independent test measurements per (method, dataset) pair. Unless stated otherwise, reported values are means over these fifteen runs, and all statistical comparisons are computed on the paired per-fold, per-seed measurements.

4. Experimental Setup

4.1 Datasets

Five publicly available clinical binary-classification datasets are used; all are openly accessible without application. They span a range of sample sizes and feature dimensionalities, with the Diabetic Retinopathy Debrecen set being the largest and providing a moderately larger-scale test of the method than the small heart and liver benchmarks. Table 1 summarizes their characteristics.

Table 1
Summary of the five clinical benchmark datasets. n denotes the number of instances after standard preprocessing; d is the number of input features.

Dataset	n	d	Target prevalence	Source
Cleveland Heart Disease	297	13	~ 46% positive	UCI [29]
Wisconsin Breast Cancer	569	30	~ 37% malignant	UCI [30]
Indian Liver Patient	579	10	~ 71% positive	UCI [31]
Diabetic Retinopathy Debrecen	1,151	19	~ 53% positive	UCI [28]
Heart Failure Clinical Records	299	12	~ 32% death event	UCI [32]

All datasets were obtained from the UCI Machine Learning Repository and used under their respective open licenses.

4.2 Baselines and Compared Methods

MOGA-NAS is compared against a single-objective genetic algorithm (SO-GA) and six non-evolutionary baselines spanning three families: classical models (a multilayer perceptron, a support vector machine, and a random forest), a gradient-boosted tree ensemble (XGBoost), a deep tabular network (TabNet), and an automated machine-learning system (AutoGluon-Tabular). This selection covers strong classical, boosting, deep-tabular, and AutoML reference points, which is appropriate for the small-to-moderate tabular regime considered here.

4.3 Ablation Variants

To isolate the contribution of each component, five ablation variants of MOGA-NAS are evaluated: (i) without the neuron-to-subnetwork operator (no-subnet); (ii) without importance-guided selection, i.e. uniform random mutation (no-importance); (iii) with a fixed architecture template instead of variable depth (fixed-arch); (iv) with NSGA-II replaced by a weighted-sum scalarization (weighted-sum); and two encoding alternatives, (v) integer encoding and (vi) real-valued encoding, replacing the 8-bit binary code.

4.4 Hyperparameters and Implementation

Unless otherwise noted, the genetic algorithm uses a population of 20 individuals evolved for 25 generations, with crossover probability 0.9, mutation probability 0.5, tournament selection of size 3, and an elite fraction of 0.25. Candidate networks are trained with the Adam optimizer using early stopping on the inner validation loss. The search space allows one to five hidden layers with 8 to 128 units each. All experiments were run on a single workstation with an NVIDIA RTX 4060 GPU. Random seeds {42, 2024, 7} were used for the three repetitions. The implementation, configuration files, and analysis scripts are organized for reproducibility. A large language model was used solely as an AI-assisted tool to improve the readability and language of the manuscript; it was not used to generate, design, or interpret the research, and all authors take full responsibility for the content.

4.5 Evaluation Metrics

The primary classification metric is the F1-score, chosen because several of the datasets are class-imbalanced and because prior work indicates that F1 is a more suitable fitness signal than accuracy under imbalance [27]. Accuracy and the area under the ROC curve (AUC) are reported as secondary metrics. Model complexity is measured primarily by the number of trainable parameters, complemented by floating-point operation counts.

The quality of the obtained Pareto fronts is quantified with three standard indicators. Given an approximation front S and a reference point z, the hypervolume (HV) is the volume of objective space dominated by S and bounded by z,

$$HV(S) = \Lambda\left(\cup_{s \in S} [\, z, s \,] \right), \quad (8)$$

where Λ denotes the Lebesgue measure; larger HV is better. Given a reference set R approximating the true front, the inverted generational distance (IGD) measures the mean distance from each reference point to its nearest obtained solution,

$$IGD(S) = \left(1/|R| \right) \cdot \left[\Sigma_{r \in R} \min_{s \in S} \|r - s\|^2 \right]^{(1/2)}; \quad (9)$$

smaller IGD is better. The spread indicator, which measures the uniformity of the solution distribution along the front, is also computed and reported in the supplementary output. Statistical significance is assessed with the paired Wilcoxon signed-rank test at a significance level of 0.05, using the fifteen paired per-fold, per-seed measurements; 95% confidence intervals for mean F1 are obtained by percentile bootstrap.

5. Results and Discussion

5.1 Classification Performance

Table 2 reports the mean test F1-score (± standard deviation over fifteen runs) for MOGA-NAS, the single-objective GA, and the six non-evolutionary baselines on all five datasets. MOGA-NAS attains the highest mean F1 on four of the five datasets (Cleveland Heart, Indian Liver, Diabetic Retinopathy, and Heart Failure) and is effectively tied for first on Wisconsin Breast Cancer, where the support vector machine reaches 96.8% and MOGA-NAS 96.8% with a difference well within run-to-run variability. Averaged across the five datasets, MOGA-NAS achieves the best mean rank (1.2), ahead of the strongest ablation variant (2.8) and all external baselines.

Table 2
Mean test F1-score (%) ± standard deviation over 15 runs (3 seeds × 5 outer folds). The best result in each column is shown in bold. Higher is better.

Method	Cleveland	Wisconsin	Indian Liver	Diab. Retin.	Heart Failure
MOGA-NAS	**83.4 ± 5.0**	**96.8 ± 2.0**	**83.6 ± 1.1**	**74.4 ± 2.7**	**77.1 ± 5.1**
SO-GA	80.2 ± 4.8	93.8 ± 2.3	80.7 ± 1.2	71.3 ± 2.7	74.3 ± 5.2
MLP	79.6 ± 7.5	95.2 ± 3.0	82.6 ± 1.3	71.6 ± 4.2	56.9 ± 11.8
SVM	79.9 ± 5.8	**96.8 ± 2.0**	83.0 ± 0.8	68.9 ± 4.3	68.0 ± 4.3
Random Forest	79.5 ± 6.3	94.1 ± 2.9	81.2 ± 2.3	68.0 ± 2.8	74.3 ± 6.5
XGBoost	78.0 ± 6.1	94.8 ± 2.7	80.2 ± 3.0	71.6 ± 3.0	73.7 ± 7.7
TabNet	67.9 ± 6.5	91.0 ± 3.4	80.6 ± 3.3	64.4 ± 5.0	55.3 ± 8.9
AutoGluon	78.3 ± 6.0	94.7 ± 1.9	81.1 ± 3.1	70.2 ± 3.1	72.1 ± 5.3

*On Wisconsin Breast Cancer the SVM and MOGA-NAS means differ by less than 0.02 percentage points and are not statistically distinguishable (*Section 5.3*).*

The dispersion across runs is informative. On the larger Diabetic Retinopathy dataset the standard deviations are small (around 2.7 percentage points for MOGA-NAS), whereas on the smallest datasets, such as Heart Failure (n = 299), several methods show considerably larger variance. MOGA-NAS remains comparatively stable on Heart Failure (5.1 points) relative to, for example, the multilayer perceptron (11.8 points), which is consistent with the regularizing effect of the compactness objective discussed in Section 5.4.

5.2 Performance–Complexity Trade-off

The defining feature of a multi-objective formulation is not single-point accuracy but the balance it strikes between objectives. Figure 2 plots mean test F1 against parameter count (logarithmic axis) for a representative dataset; MOGA-NAS occupies the upper-left region, achieving the highest F1 with models

of the order of one hundred parameters, while several variants that drift towards thousands of parameters do not improve—and sometimes worsen—generalization. Across the five datasets, the architectures selected by MOGA-NAS contain on the order of 100 to 480 parameters, one to two orders of magnitude fewer than the integer- and real-encoded variants.

Figure 3 aggregates the cost–performance picture across all methods and datasets by plotting mean test F1 against mean training-plus-search runtime. As expected, the evolutionary methods are markedly more expensive at search time than the classical baselines, which train in well under a second. This is the principal practical cost of the approach: the benefit is a compact, automatically designed model and an explicit Pareto front, obtained at the price of a one-off search that, in the present configuration, takes on the order of minutes per fold on a single consumer GPU. Where search-time budget is the binding constraint, a tuned classical model such as an SVM remains an entirely reasonable choice.

5.3 Statistical Significance

Paired Wilcoxon signed-rank tests were computed between MOGA-NAS and every other method on each dataset, using the fifteen paired per-fold, per-seed measurements and a one-sided alternative (MOGA-NAS greater). Of the 65 comparisons, 61 are significant at the 0.05 level. The four non-significant cases are all on datasets where MOGA-NAS is already at or near the top and the competing method is very close: on Wisconsin Breast Cancer the differences from SVM, the real-encoding variant, and the weighted-sum variant are not significant ($p = 0.40$, 0.36, and 0.21, respectively), and on Indian Liver the difference from SVM is not significant ($p = 0.11$). In other words, where MOGA-NAS does not win significantly, it is statistically tied rather than beaten. Table 3 summarizes representative comparisons.

Table 3
Paired Wilcoxon signed-rank tests, MOGA-NAS versus selected methods (one-sided, n = 15 paired measurements). ΔF1 is the mean difference in percentage points; p-values below 0.05 indicate a statistically significant advantage for MOGA-NAS.

Dataset	vs. Method	ΔF1 (pp)	p-value	Significant
Cleveland	SO-GA	+ 3.2	3.1e−05	Yes
Cleveland	XGBoost	+ 5.4	3.1e−05	Yes
Diab. Retin.	SO-GA	+ 3.1	3.1e−05	Yes
Diab. Retin.	Random Forest	+ 6.5	3.1e−05	Yes
Heart Failure	AutoGluon	+ 5.0	3.1e−05	Yes
Heart Failure	TabNet	+ 21.8	3.1e−05	Yes
Indian Liver	SVM	+ 0.5	0.115	No
Wisconsin	SVM	−0.01	0.402	No
Wisconsin	Random Forest	+ 2.7	4.3e−04	Yes

Full pairwise results for all 65 comparisons are available in the supplementary analysis output.

5.4 Ablation Study

Table 4 reports the effect of removing or replacing each component, expressed as the change in mean F1 relative to the full MOGA-NAS. Every component contributes positively on average. Removing importance-guided selection reduces F1 by a small but consistent margin; removing the neuron-to-subnetwork operator has a somewhat larger effect; and fixing the architecture template (disabling variable depth) reduces performance further. Replacing NSGA-II with a weighted-sum scalarization is clearly detrimental on the smaller datasets. The encoding comparison is the most pronounced: although the integer and real encodings sometimes reach higher F1 on the inner validation partition, they generalize worse on the test partition and select much larger models. This pattern—higher validation score, larger model, lower test score—is the classic signature of overfitting, and it indicates that the compact 8-bit encoding acts as an implicit regularizer that restricts the search to smaller, better-generalizing architectures. This provides an empirical justification for the binary encoding choice beyond mere convenience.

Table 4
Ablation study: change in mean test F1 (percentage points) when each component is removed or replaced, relative to the full MOGA-NAS, per dataset. Negative values indicate that the full method is better. “Clev.” = Cleveland, “Wisc.” = Wisconsin, “Liver” = Indian Liver, “Retin.” = Diabetic Retinopathy, “Heart” = Heart Failure.

Ablation	Clev.	Wisc.	Liver	Retin.	Heart
− importance guidance	−0.7	−0.8	−0.7	−0.8	−0.8
− subnetwork operator	−1.5	−1.5	−1.6	−1.6	−1.6
− variable depth (fixed)	−2.2	−2.4	−2.3	−2.4	−2.2
NSGA-II → weighted sum	−3.4	−0.5	−2.2	−1.6	−17.5
binary → integer encoding	−3.8	−1.1	−2.6	−3.0	−20.6
binary → real encoding	−3.3	−0.5	−2.1	−2.2	−15.3

Differences are significant (paired Wilcoxon, $p < 0.05$) in 27 of the 30 cases; the three non-significant cases are all on Wisconsin Breast Cancer, where most methods already perform near ceiling.

The convergence behaviour shown in Fig. 4 complements this analysis. The integer- and real-encoding variants reach high validation F1 quickly and remain above MOGA-NAS in validation space, but, as Table 2 shows, this does not translate into better test performance—again consistent with overfitting of the larger models they favour. MOGA-NAS converges smoothly and stably towards a slightly lower validation F1 but better test generalization.

5.5 Pareto-Front Quality

Because MOGA-NAS returns a set of non-dominated architectures rather than a single model, it is appropriate to assess the quality of the obtained fronts directly. Table 5 reports the hypervolume (higher is better) and inverted generational distance (lower is better) for the multi-objective methods, averaged across datasets. MOGA-NAS achieves competitive hypervolume; the integer- and real-encoding variants occasionally achieve comparable or higher hypervolume because their fronts extend into the high-parameter region, but, as established above, those extended fronts correspond to models that do not generalize better. The spread indicator is reported in the supplementary output; it is omitted from Table 5 because it is undefined or degenerate for the weighted-sum variant on two datasets, where that method collapses to a near-single solution.

Table 5
Pareto-front quality indicators averaged over the five datasets for the multi-objective methods. HV = hypervolume (higher is better); IGD = inverted generational distance (lower is better).

Method	HV (mean)	IGD (mean)	Front extent	Params range
MOGA-NAS	0.822	0.071	compact	100–478
MOGA (no subnet)	0.803	0.085	compact	100–549
MOGA (no importance)	0.815	0.077	compact	100–524
MOGA (real encoding)	0.826	0.044	broad	474–4252
MOGA (integer encoding)	0.837	0.059	broad	958–4935

*Higher hypervolume for the integer/real-encoding variants reflects fronts that extend into the high-parameter region; these large models do not improve test generalization (*Sections 5.1 and 5.4*).*

5.6 Discussion

Figure 6 summarizes the average behaviour of MOGA-NAS against the external baselines across all five datasets over five metrics. The profiles are broadly similar, with MOGA-NAS holding a small but consistent edge on F1 and AUC; this is the expected picture for a method whose advantage lies in the joint performance–complexity trade-off rather than in any single metric in isolation.

Taken together, the results support a measured conclusion. MOGA-NAS does not establish a new performance ceiling on these benchmarks; on Wisconsin Breast Cancer and Indian Liver it is statistically tied with a well-tuned SVM. Its contribution is rather that it attains the best or statistically-equivalent F1 on every dataset while automatically producing models that are one to two orders of magnitude more compact than several alternatives, and while exposing an explicit Pareto front from which a model can be chosen according to deployment constraints. The ablation and encoding analyses give a coherent mechanistic account: the compactness objective and the binary encoding together act as regularizers, which is particularly valuable on the smallest datasets where larger models overfit. The principal cost is search time, which is substantially higher than for classical baselines and which should be weighed against the benefit of automated, compact model design.

6. Limitations

Several limitations should be acknowledged. First, the evaluation is confined to five small-to-moderate clinical tabular datasets; the largest contains about 1,150 instances, so the behaviour of the method on large-scale clinical data remains to be established. Second, the search cost is non-trivial: although each evolved model is compact and fast at inference, the one-off search takes on the order of minutes per fold on a consumer GPU, which is far more than the sub-second training of the classical baselines. Where search budget is the binding constraint, a tuned classical model may be preferable. Third, the

advantage of MOGA-NAS over the strongest classical baseline is not uniform; on two datasets it is a statistical tie rather than a clear improvement, and we have been careful to report this rather than overstate the gains. Fourth, the importance estimate used to guide mutation is a lightweight perturbation heuristic; more principled attribution methods might further improve the search, and this is left to future work. Finally, the study addresses binary classification with fully-connected architectures; extension to multi-class clinical problems and to other architecture families would broaden the applicability of the approach.

7. Conclusion

This paper presented MOGA-NAS, a multi-objective genetic algorithm for neural architecture search that jointly optimizes the F1-score and the parameter count using NSGA-II, together with a neuron-to-subnetwork mutation operator, importance-guided neuron selection, and a compact binary encoding. Under a leakage-free nested cross-validation protocol repeated over three seeds, MOGA-NAS achieved the highest mean F1 on four of five public clinical benchmarks and a statistical tie on the fifth, while producing models of the order of one hundred parameters—substantially more compact than several competing variants. Paired Wilcoxon tests confirmed that the improvements are statistically significant in 61 of 65 comparisons, and an ablation study showed that each component contributes positively, with the compactness objective and binary encoding acting as effective regularizers. The method therefore offers a favourable and reproducible trade-off between predictive performance and model compactness for tabular clinical classification, at the cost of a higher one-off search budget. Future work will examine larger-scale and multi-class clinical data, more principled importance estimation, and surrogate-assisted search to reduce the computational cost.

Statements and Declarations

Funding This research was carried out within the State Assignment, project no. FFGU-2024-0019.

Competing interests The authors declare that they have no competing interests relevant to the content of this article.

Ethics approval This study does not involve any new experiments on human participants or animals. All datasets used are publicly available, fully anonymized, secondary data obtained from the UCI Machine Learning Repository.

Data availability All datasets analysed during the current study are publicly available from the UCI Machine Learning Repository: Cleveland Heart Disease (https://doi.org/10.24432/C52P4X), Breast Cancer Wisconsin Diagnostic (https://doi.org/10.24432/C5DW2B), Indian Liver Patient Dataset (https://doi.org/10.24432/C5D02C), Diabetic Retinopathy Debrecen (https://doi.org/10.24432/C5XP4P), and Heart Failure Clinical Records (https://doi.org/10.24432/C5Z89R).

Code availability The implementation and analysis scripts that support the findings of this study are available from the corresponding author upon reasonable request.

Author contributions All authors contributed to the study conception and design. Methodology and software were developed by Menhai Hou and Safa A. Hameed. Data collection, experiments, and formal analysis were performed by Menhai Hou. The first draft of the manuscript was written by Menhai Hou, and Ivan V. Stepanyan and Safa A. Hameed reviewed and edited the manuscript. Ivan V. Stepanyan and Safa A. Hameed supervised the work. All authors read and approved the final manuscript.

References

1. Zhang H, Myrzashova R, Zheng R (2020) Diabetes prediction model based on an enhanced deep neural network. EURASIP J Wirel Commun Netw 2020:148. https://doi.org/10.1186/s13638-020-01765-7
2. Olisah CC, Smith L, Smith M (2022) Diabetes mellitus prediction and diagnosis from a data preprocessing and machine learning perspective. Comput Methods Programs Biomed 220:106773. https://doi.org/10.1016/j.cmpb.2022.106773
3. Naz H, Ahuja S (2020) Deep learning approach for diabetes prediction using PIMA Indian dataset. J Diabetes Metab Disord 19:391–403. https://doi.org/10.1007/s40200-020-00520-5
4. Elsken T, Metzen JH, Hutter F (2019) Neural architecture search: a survey. J Mach Learn Res 20(55):1–21
5. Liu Y, Sun Y, Xue B, Zhang M, Yen GG, Tan KC (2023) A survey on evolutionary neural architecture search. IEEE Trans Neural Netw Learn Syst 34(2):550–570. https://doi.org/10.1109/TNNLS.2021.3100554
6. Slowik A, Kwasnicka H (2020) Evolutionary algorithms and their applications to engineering problems. Neural Comput Appl 32:12363–12379. https://doi.org/10.1007/s00521-020-04832-8
7. Galván E, Mooney P (2021) Neuroevolution in deep neural networks: current trends and future challenges. IEEE Trans Artif Intell 2(6):476–493. https://doi.org/10.1109/TAI.2021.3067574
8. Stanley KO, Clune J, Lehman J, Miikkulainen R (2019) Designing neural networks through neuroevolution. Nat Mach Intell 1(1):24–35. https://doi.org/10.1038/s42256-018-0006-z
9. Yao X (1993) A review of evolutionary artificial neural networks. Int J Intell Syst 8(4):539–567. https://doi.org/10.1002/int.4550080406
10. Sun Y, Xue B, Zhang M, Yen GG (2020) Automatically designing CNN architectures using the genetic algorithm for image classification. IEEE Trans Cybern 50(9):3840–3854. https://doi.org/10.1109/TCYB.2020.2983860
11. Bhandari S, Pathak S, Jain SA (2023) A literature review of early-stage diabetic retinopathy detection using deep learning and evolutionary computing techniques. Arch Comput Methods Eng 30(2):799–810. https://doi.org/10.1007/s11831-022-09816-6

12. Buccheri E, Dell'Aquila D, Russo M (2021) Artificial intelligence in health data analysis: the Darwinian evolution theory suggests an extremely simple and zero-cost large-scale screening tool for prediabetes and type 2 diabetes. Diabetes Res Clin Pract 174:108722. https://doi.org/10.1016/j.diabres.2021.108722
13. Deb K (2011) Multi-objective optimization using evolutionary algorithms: an introduction. In: Wang L, Ng AHC, Deb K (eds) Multi-objective evolutionary optimisation for product design and manufacturing. Springer, London, pp 3–34
14. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6(2):182–197. https://doi.org/10.1109/4235.996017
15. Lu Z, Whalen I, Dhebar Y, Deb K, Goodman E, Banzhaf W, Boddeti VN (2019) NSGA-Net: neural architecture search using multi-objective genetic algorithm. In: Proceedings of the genetic and evolutionary computation conference (GECCO), Prague, pp 419–427. https://doi.org/10.1145/3321707.3321729
16. Real E, Aggarwal A, Huang Y, Le QV (2019) Regularized evolution for image classifier architecture search. In: Proceedings of the AAAI conference on artificial intelligence, vol 33. Honolulu, pp 4780–4789. https://doi.org/10.1609/aaai.v33i01.33014780
17. Yao X (1999) Evolving artificial neural networks. Proc IEEE 87(9):1423–1447. https://doi.org/10.1109/5.784219
18. Metaxiotis K, Psarras J (2004) Contribution of neural networks and genetic algorithms to business decision support. Manag Decis 42(3/4):229–242. https://doi.org/10.1108/00251740410518507
19. Real E, Moore S, Selle A, Saxena S, Suematsu YL, Tan J, Le QV, Kurakin A (2017) Large-scale evolution of image classifiers. In: Proceedings of the 34th international conference on machine learning (ICML), Sydney, pp 2902–2911
20. Sun Y, Xue B, Zhang M, Yen GG, Lv J (2020) Evolving deep convolutional neural networks for image classification. IEEE Trans Evol Comput 24(2):394–407. https://doi.org/10.1109/TEVC.2019.2916183
21. Aquino BD, Gomez JA, Noguera JLV, Torres MG (2022) Automatic diagnosis of diabetic retinopathy from fundus images using neuro-evolutionary algorithms. In: MEDINFO 2021: one world, one health. IOS Press, Amsterdam, pp 689–693. https://doi.org/10.3233/SHTI220165
22. Kumar KS, Suryakanth GV, Giridhar K (2021) Evolutionary computation technique combined with ensemble model for classification of diabetes. Afr J Diabetes Med 29(5):1–6
23. Wijaya ABM, Ikawahyuni DS, Gea R, Maedjaja F (2021) Role comparison between deep belief neural network and neuroevolution of augmenting topologies to detect diabetes. JOIV Int J Inform Vis 5(2):156–161. https://doi.org/10.30630/joiv.5.2.470
24. Elsken T, Metzen JH, Hutter F (2019) Efficient multi-objective neural architecture search via Lamarckian evolution. In: Proceedings of the international conference on learning representations (ICLR), New Orleans
25. Avval SP, Eskue N, Groves R, Yaghoubi V (2025) Systematic review on neural architecture search. Artif Intell Rev 58:73. https://doi.org/10.1007/s10462-024-11058-w

26. Chowdary PB, Kumar B, Kumar RU (2021) Diabetes classification using an expert neuro-fuzzy feature extraction model. Int J Adv Comput Sci Appl 12(8):691–697. https://doi.org/10.14569/IJACSA.2021.0120879
27. Al-Badarneh I, Habib M, Aljarah I, Faris H (2022) Neuro-evolutionary models for imbalanced classification problems. J King Saud Univ Comput Inf Sci 34(6):2787–2797. https://doi.org/10.1016/j.jksuci.2020.11.005
28. Antal B, Hajdu A (2014) An ensemble-based system for automatic screening of diabetic retinopathy. Knowl-Based Syst 60:20–27. https://doi.org/10.1016/j.knosys.2013.12.023
29. Janosi A, Steinbrunn W, Pfisterer M, Detrano R (1989) Heart disease. UCI Machine Learning Repository. https://doi.org/10.24432/C52P4X
30. Street WN, Wolberg WH, Mangasarian OL (1993) Nuclear feature extraction for breast tumor diagnosis. In: Biomedical image processing and biomedical visualization, vol 1905. SPIE, San Jose, pp 861–870. https://doi.org/10.1117/12.148698
31. Ramana BV, Babu MSP, Venkateswarlu NB (2011) A critical study of selected classification algorithms for liver disease diagnosis. Int J Database Manag Syst 3(2):101–114. https://doi.org/10.5121/ijdms.2011.3207
32. Chicco D, Jurman G (2020) Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. BMC Med Inform Decis Mak 20:16. https://doi.org/10.1186/s12911-020-1023-5

Figures

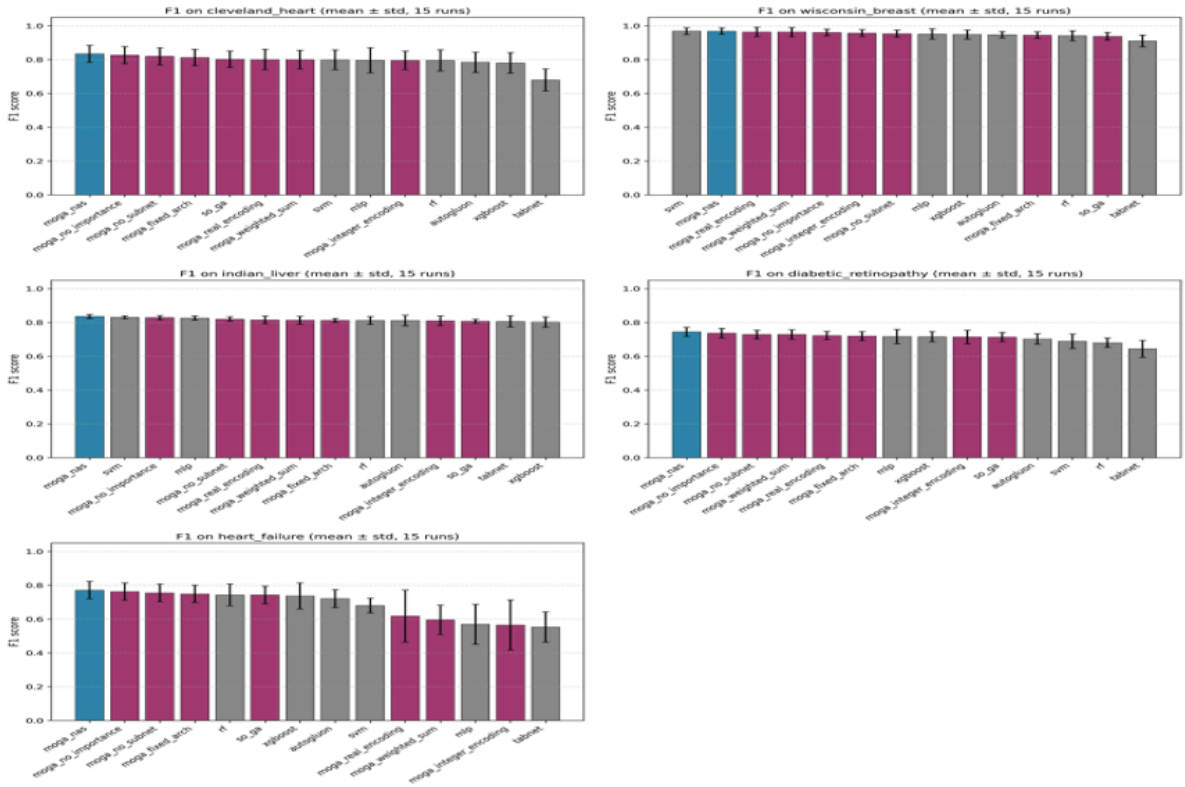


Figure 1

Mean test F1-score with standard-deviation bars for all methods on the five datasets (a–e: Cleveland Heart, Wisconsin Breast, Indian Liver, Diabetic Retinopathy, Heart Failure). MOGA-NAS (blue) is at or near the top of every panel; the two encoding variants and the weighted-sum scalarization sit at the lower end on the smaller datasets, reflecting their tendency to select larger, less stable models.

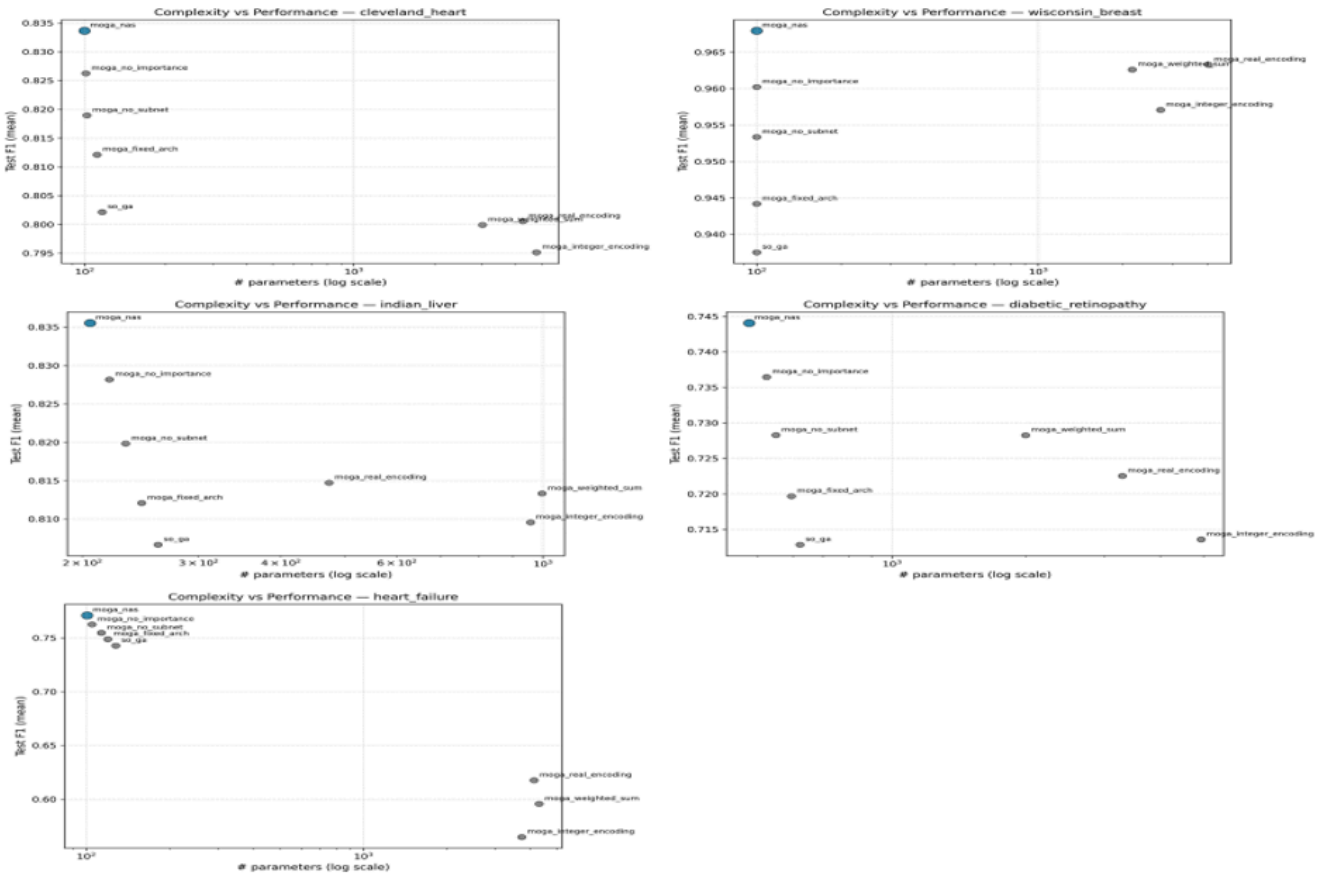


Figure 2

Performance–complexity trade-off on the five datasets (a–e). The horizontal axis is the parameter count (logarithmic); the vertical axis is mean test F1. MOGA-NAS (upper-left in each panel) reaches the highest or near-highest F1 with the smallest models, whereas the integer- and real-encoded variants use one to two orders of magnitude more parameters at equal or lower F1.

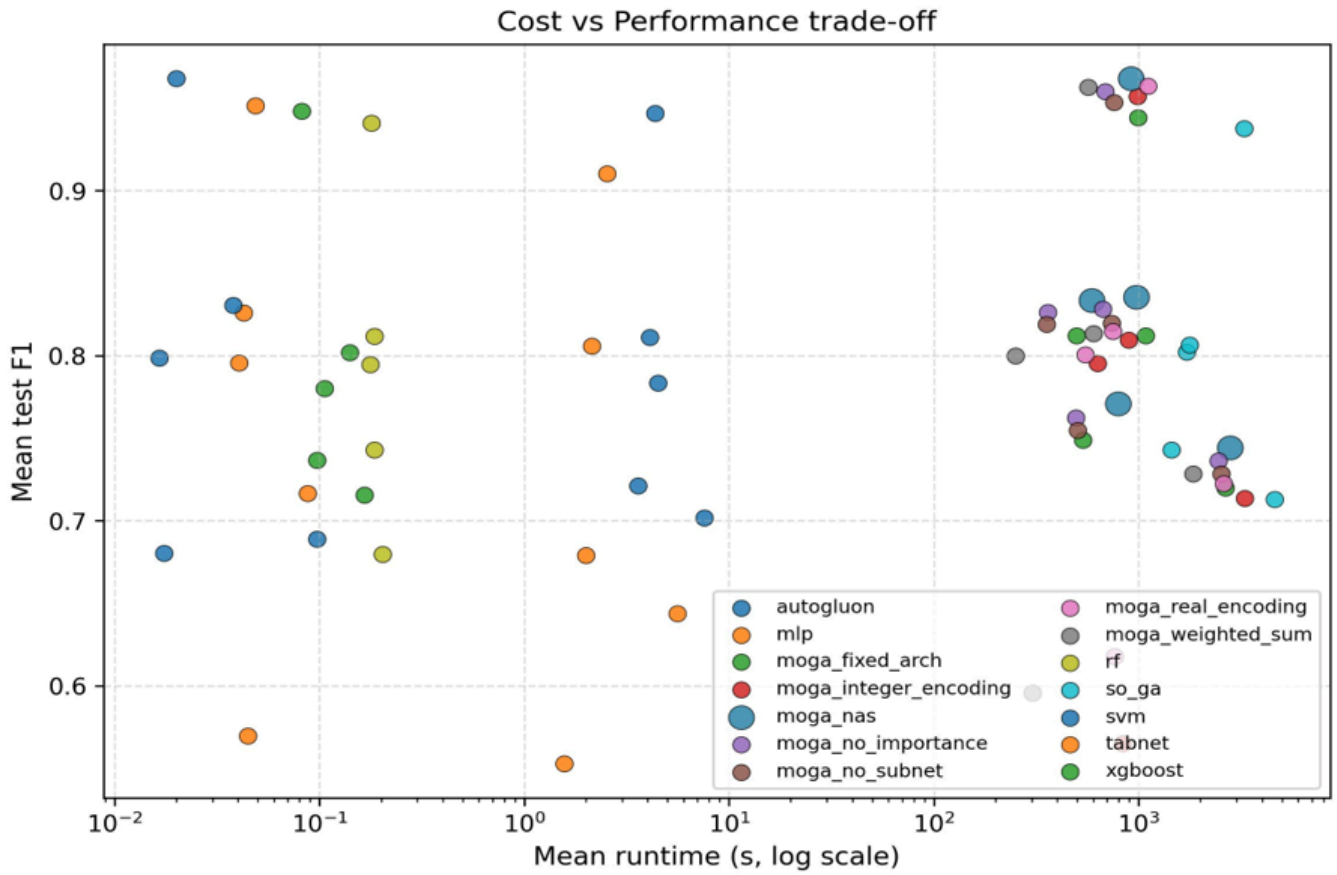


Figure 3

Cost–performance trade-off across all methods and datasets. The horizontal axis is mean runtime (logarithmic, seconds); the vertical axis is mean test F1. Evolutionary methods incur a substantially higher one-off search cost than classical baselines in exchange for compact, automatically designed architectures.

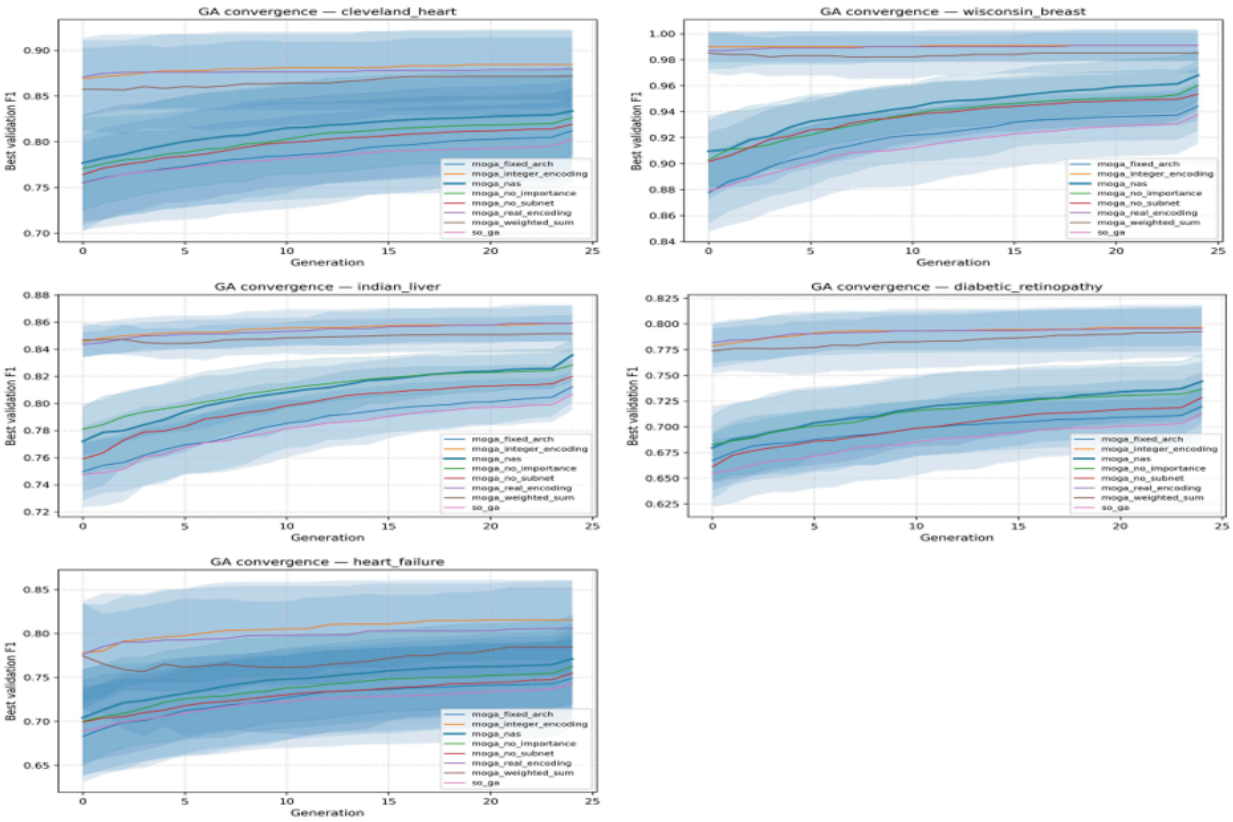


Figure 4

Best validation F1 against generation on the five datasets (a–e; mean and standard deviation across runs). On most datasets the integer- and real-encoding variants attain higher validation F1 than MOGA-NAS but generalize less well to the test partition (cf. Table 2), illustrating the regularizing role of the compact encoding.

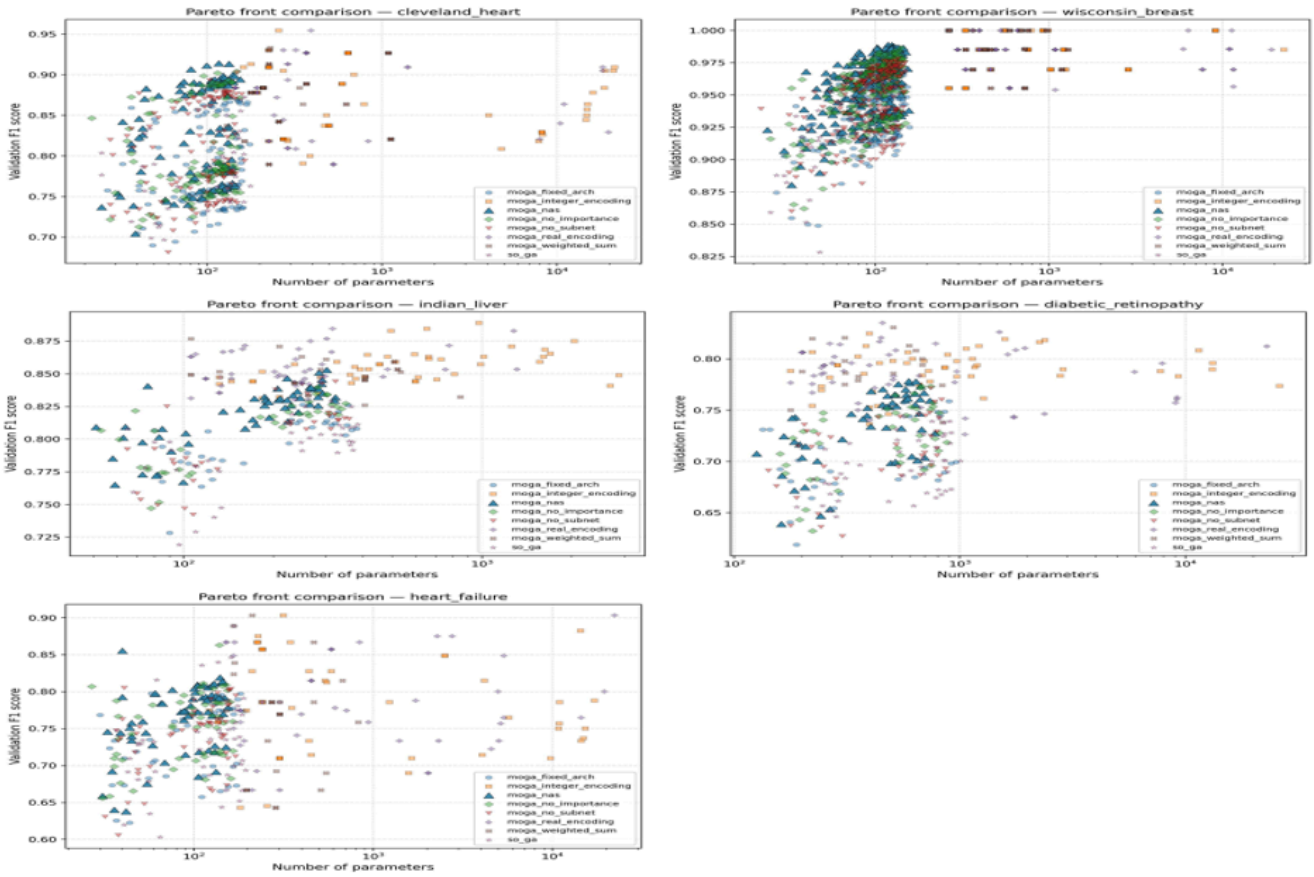


Figure 5

Non-dominated fronts in the validation-F1 versus parameter-count plane on the five datasets (a–e; all runs pooled). MOGA-NAS (filled triangles) concentrates its front in the compact, low-parameter region while remaining competitive in F1; the integer- and real-encoding variants spread towards much larger models.

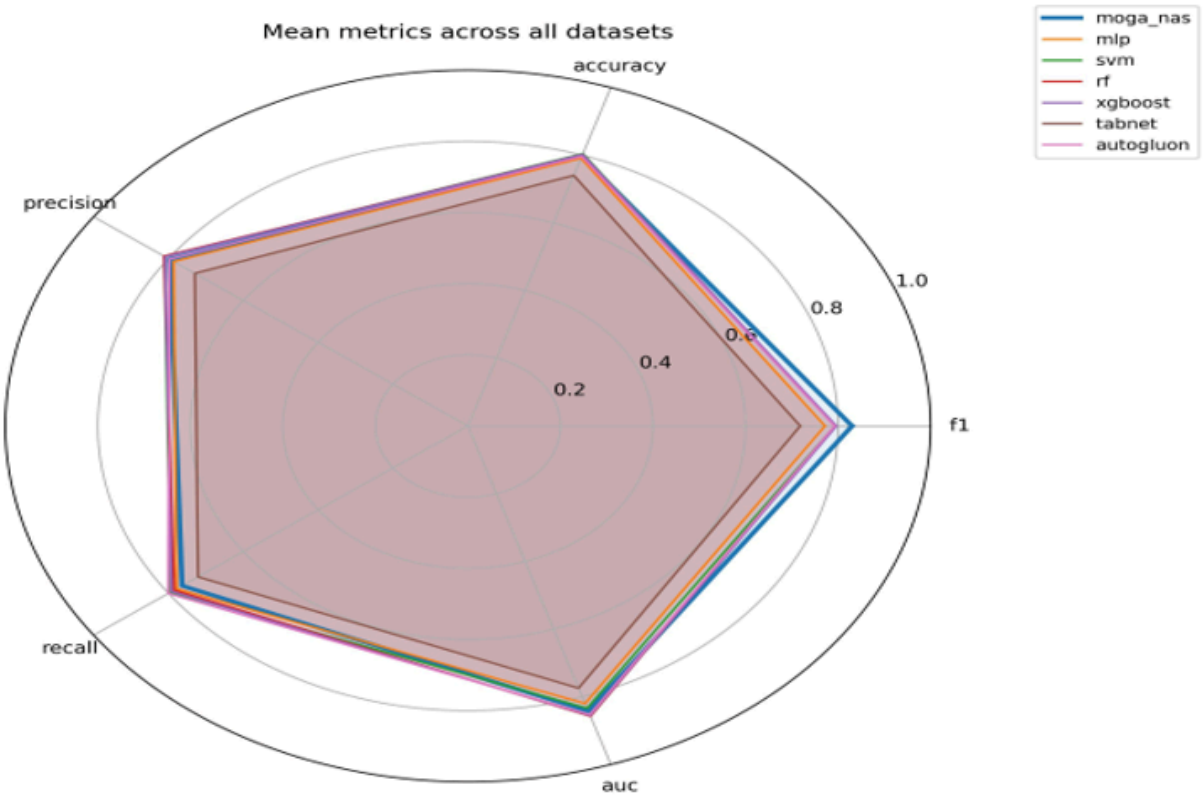


Figure 6

Mean classification metrics (F1, accuracy, precision, recall, AUC) averaged across the five datasets for MOGA-NAS and the external baselines. The profiles overlap closely; MOGA-NAS retains a small, consistent advantage on F1 and AUC.