TABBENCH-BIO: A LIVING BENCHMARK FOR MACHINE LEARNING ON HIGH-DIMENSIONAL BIOMEDICAL TABLES

Anonymous authors
Paper under double-blind review

ABSTRACT

Biomedical tables often combine thousands of measured variables with only tens or hundreds of labelled samples, a regime that is poorly represented in general-purpose tabular benchmarks. We introduce TabBench-Bio, a living and interactive benchmark of 43 biomedical datasets spanning multiple domains. Under a shared cross-validation protocol, we compare classical estimators, neural networks, and tabular foundation models across 28 feature-by-sample operating points. At the reference cell of 10,000 features and 100 training samples, RealTabPFN 2.5 has the highest point estimate, closely followed by TabPFN 3 and Logistic Regression. A paired bootstrap over the target pool separates RealTabPFN 2.5 from TabPFN 3 by 87 Elo (95% interval [43, 129]). Tabular foundation models generally occupy the leading ranks, while the strongest configuration depends on the operating point and biomedical modality. The AutoML framework AutoGluon, using its one-hour "extreme" preset, is configured as a separate resource-intensive reference and reported here at the reference and full cell. Fold-level predictions, run status, and deterministic aggregations make every reported result reproducible and reusable. The benchmark is open to contributions of new biomedical datasets. The interactive leaderboard is available at https://anonymous_iclr.codeberg.page/TabBench-Bio-Anonymous/.

1 INTRODUCTION

Comparative evaluation of machine learning on biomedical tables remains heterogeneous, as individual studies typically select limited cohorts, assays, and estimators. Differences in preprocessing and validation make results difficult to compare, a problem exacerbated by the arrival of tabular foundation models (TFMs) (Müller et al., 2022), whose priors may help when data is scarce but whose computational behaviour differs from linear models and boosted trees. Furthermore, most comparisons report only one operating point per dataset, obscuring whether a model is robust to feature width or cohort size in high-dimensional, low-sample-size (HDLSS) settings. We therefore treat a controlled feature-by-sample grid as a primary part of the benchmark definition.

General tabular benchmarks like TabArena (Erickson et al., 2025) and TALENT (Liu et al., 2025) provide a foundation by curating datasets, implementations, and leaderboards, while Grinsztajn et al. (2022) identify when tree-based models remain strong. However, these focus primarily on conventional independent and identically distributed (IID) tasks and provide limited evidence on the HDLSS geometries common in molecular biomedicine.

In molecular biomedicine, gene-expression, metagenomic, and methylation profiles often expose thousands of variables for only tens or hundreds of samples. In this HDLSS regime, the feature-to-sample ratio affects both statistical estimation and model feasibility. Classical estimators face irrelevant measurements, neural networks lack sufficient supervision, and TFMs may exceed their original feature limits. For instance, TabPFN v2's performance on small-to-medium tables (Hollmann et al., 2025) does not establish how its prior behaves when feature width and cohort size are varied independently.

RamanBench (Koddenbrock et al., 2026) highlights the value of domain-focused HDLSS benchmarks. Raman spectra have a physically ordered wavenumber axis, whereas the gene-expression and

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

metagenomic tables considered here do not share an analogous column ordering. Biomedical HDLSS data therefore require a benchmark spanning multiple measurement processes rather than treating one assay as a proxy for the whole regime.

We introduce TabBench-Bio, a living benchmark combining a versioned registry, common execution interface, and deterministic reporting. Following TabArena's maintained perspective and RamanBench's domain-specific motivation, we evaluate a 4-by-7 grid (28 operating points) with shared cross-validation folds and nested feature caps. This ensures the held-out population remains fixed, making robustness across data geometries a primary outcome. Our main contributions are as follows:

- a versioned registry of 43 biomedical datasets spanning heterogeneous molecular and sequence-derived representations, all evaluated under the shared benchmark protocol;
- a common implementation-level comparison of 19 predictive model configurations and a constant baseline, with AutoGluon configured as a separate resource-intensive AutoML reference at two prespecified operating points;
- a leak-resistant 4-by-7 grid with fold-aligned evaluation, explicit failure semantics, and cell-conditional Bradley-Terry rankings;
- reusable fold-level predictions, run status, resource measurements, deterministic aggregations, and an interactive leaderboard.

2 Related work

Earlier tabular studies established that the relative performance of trees and neural networks depends on dataset scale, feature type, and tuning protocol (Grinsztajn et al., 2022; Somvanshi et al., 2026). TALENT (Liu et al., 2025) and TabArena (Erickson et al., 2025) provide frameworks for consistent evaluation and benchmark governance, with TabArena emphasizing a living leaderboard and reusable predictions. TabBench-Bio adopts this versioned structure but focuses on biomedical HDLSS tasks and the robustness of fixed configurations across resource regimes, rather than extensively tuned performance. This domain shift is critical: TabArena's findings on foundation models and tuning (Erickson et al., 2025) describe general-purpose tasks and may not carry over to biomedical HDLSS data.

Gradient-boosted trees (XGBoost, LightGBM, CatBoost) remain essential baselines for heterogeneous tables (Chen & Guestrin, 2016; Ke et al., 2017; Prokhorenkova et al., 2018). Neural tabular learning has evolved from generic MLPs to specialised architectures like RealMLP (Holzmüller et al., 2024) and TabM (Gorishniy et al., 2025). Tabular foundation models (TFMs) for in-context learning include RealTabPFN v2, 2.5 for small-data prediction, TabPFN-Wide for high feature counts, TabPFN 3, TabDPT, and Mitra (Garg et al., 2025; Grinsztajn et al., 2025; 2026; Kolberg et al., 2025; Zhang et al., 2025; Ma et al., 2025), as well as TabICLv2 (Qu et al., 2026), the successor to TabICL (Qu et al., 2025), and TabFM (Kong & Das, 2026). Because these models make different assumptions about how evidence scales with rows and columns, TabBench-Bio's two-dimensional grid makes these dependencies visible.

RamanBench (Koddenbrock et al., 2026) provides a methodological precedent through its live leaderboard for Raman spectroscopy. Architectures using spectral locality require an appropriate representation or adaptation before application to omics tables without an analogous column ordering. TabBench-Bio examines whether general-purpose tabular learners persist across heterogeneous biomedical representations that share no physical column axis.

Various resources provide biomedical data: OpenML (Vanschoren et al., 2014), GEO and TCGA for molecular profiles (Edgar et al., 2002; Weinstein et al., 2013), MGnify for microbiome studies (Mitchell et al., 2020), TDC for drug discovery (Huang et al., 2022), and ChEMBL for bioactivity (Zdrazil et al., 2024). Genomic-prediction panels also link SNPs to quantitative traits (Azodi et al., 2019), while protein-language-model benchmarks provide sequence-derived representations (Lin et al., 2023; Thumuluri et al., 2022; Xu et al., 2022). TabBench-Bio unifies these sources in one registry, evaluating them under a shared protocol for splitting, fitting, scoring, and aggregation.

3 TABBENCH-BIO

TabBench-Bio evaluates supervised classification and regression on static, pre-extracted numerical representations of biomedical samples in the HDLSS regime.

Using default configurations and fixed validation folds, we ask how tabular learners behave as the number of labelled samples and the feature width change. Our current scope excludes survival analysis, longitudinal modelling, multimodal integration, external-site validation, and supervised feature selection. Each would change the prediction unit, endpoint, or source of supervision addressed by the current protocol.

3.1 DATASET SCOPE AND SELECTION

The registry contains 43 datasets: 30 classification and 13 regression tasks. MGnify and OpenML contribute nine and ten datasets, respectively; the remainder come from TCGA, GEO, TDC, ChEMBL, Dryad/Zenodo, and curated or locally derived releases. The datasets span 7 modality groups with different measurement processes, sparsity, and dependence structures, allowing comparisons across biological representations rather than a single assay. Appendix G lists the full registry.

The collection includes six molecular-property tasks from TDC, two single-protein pChEMBL regression endpoints from ChEMBL 37, four crop genomic-prediction tasks, four DNA methylation cohorts from GEO, two protein-embedding tasks, and a metagenomic marker-profile cohort for liver cirrhosis. For the ChEMBL tasks, we use exact, unflagged activity measurements (Zdrazil et al., 2024), ECFP4 fingerprints (Rogers & Hahn, 2010), and Bemis-Murcko scaffolds as split groups (Bemis & Murcko, 1996). The crop tasks pair SNP panels for maize, rice, soybean, and switchgrass with flowering-time, yield, or height phenotypes (Azodi et al., 2019); the methylation tasks retain their source-study endpoints (Hannum et al., 2013; Liu et al., 2013). Three locally constructed datasets use mean-pooled embeddings: Evo 2 for BRCA1 variant sequences and ESM-2 for DeepLoc 2.0 fungal proteins and PEER TEM-1 mutants (Brixi et al., 2026; Findlay et al., 2018; Lin et al., 2023; Thumuluri et al., 2022; Xu et al., 2022). They use biological locus, published homology partition, and mutation position, respectively, as cross-validation groups (Appendix G.1).

We combined automated screening with manual curation. Automated preselection generally required at least 70 samples and 1,000 features, non-synthetic data, and an open or reusable licence. We tested these candidates for predictive signal using a random forest with 100 label permutations, requiring $p < 0.05$ and a cross-validated score of at least 0.10 adjusted balanced accuracy for classification or 0.05 R^2 for regression; we excluded near-perfect scores above 0.98 as trivial. Manual additions were selected for relevance and quality. We checked the remaining candidates for duplicate observations, missing or constant features, inconsistent representations, outliers, potential target leakage, batch-target association, and class balance. Final review covered each target, prediction unit, licence, group identifiers, and reproducible table construction. The registry retains stable identifiers and the reviewed metadata. Before constructing folds, we remove classes with fewer than ten samples, so the benchmark does not assess rare-class recognition.

3.2 MODELS AND FITTING REGIME

The peer comparison comprises 19 predictive configurations: L2-regularised Logistic Regression for classification and Ridge Regression for regression; nearest neighbours; random forests and extra trees; CatBoost, LightGBM, and XGBoost; a standard MLP, RealMLP, and TabM; and 9 TFMs, including RealTabPFN v2, 2.5, TabPFN 3, two TabPFN-Wide configurations, Mitra, TabDPT, TabICLv2, and TabFM (Garg et al., 2025; Grinsztajn et al., 2025; 2026; Kolberg et al., 2025; Zhang et al., 2025; Ma et al., 2025; Qu et al., 2026; Kong & Das, 2026). We fit a constant predictor as an explicit baseline and use its score to impute out-of-memory or time-limit failures. The benchmark exposes all models through a common adapter built on the AutoGluon tabular interface (Erickson et al., 2020).

For each peer, we use the default configuration exposed by the registered AutoGluon model adapter, without per-dataset hyperparameter search. Benchmark-level bagging, stacking, and ensembling are disabled. Results concern these fixed configurations; the effects of bagging and hyperparameter tuning are not evaluated.

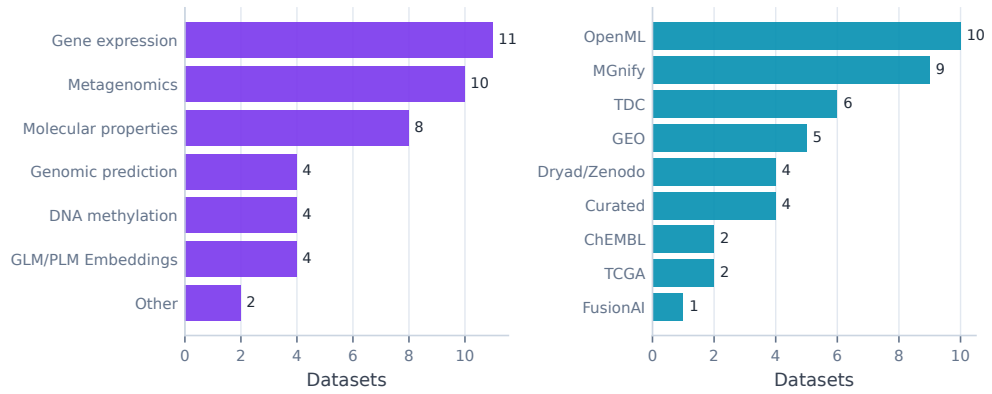

Figure 1: Benchmark dataset counts by modality group and source, generated from the versioned registry.

Appendix A lists the principal configuration parameters for every model; values not listed remain at the corresponding library defaults.

The benchmark measures how reliable each model is as a reusable default rather than the best performance obtainable after bespoke tuning. Fitting and prediction are each subject to the same external timeout of 1 hour. Each model fit was restricted to 32 threads. All GPU-based models, including the native AutoGluon Extreme reference, were evaluated on NVIDIA L40S GPUs with 48 GB of device memory, with one fit per GPU. CPU-only models were evaluated on CPU nodes. All models within a cell used the same frozen data splits and model configurations. GPU type and available memory were recorded with each run. The reported results therefore reflect the specified memory and wall-clock budgets. When prediction exhausted GPU memory, we retried the same held-out fold in smaller query batches, starting with the full fold and repeatedly halving the batch down to one row if necessary. Classification labels and probabilities were handled independently and concatenated in the original row order. The fitted model and held-out rows were retained, and the original timeout continued to apply. Runs that exhausted memory after the applicable retries were recorded as failures. We evaluated AutoGluon's native Extreme preset at two prespecified points: the 10,000-feature, 100-sample reference cell and the full-feature, full-sample cell (Figure A2). Its one-hour budget made evaluation throughout the grid computationally prohibitive. The results below use the reference cell. We report it as a resource-intensive one-hour AutoML reference rather than a ranked peer.

3.3 Operating-point grid

The grid separates the effects of feature width and labelled cohort size. Its budgets are

$$p \in \{2{,}000, 10{,}000, 25{,}000, \text{full}\}, \qquad n \in \{20, 50, 100, 200, 500, 1{,}000, \text{full}\},$$

leading to 28 different combinations. The prespecified reference cell uses 10,000 features and 100 training samples. Within each cell, all models share the same folds and training rows; the evaluation unit is a fold-target-model combination.

Sample budgeting affects only the training partition; all budgets share the same held-out fold. Regression uses seeded random subsampling, whereas classification allocates the requested budget across classes and samples within each class using the same seed. The number of training rows selected from each class is recalculated at each budget, thus a larger training subset may therefore omit rows included in a smaller subset. Data-dependent feature filtering is fitted separately on each budgeted training set. Sample-budget comparisons therefore characterize the complete budgeted learning procedure rather than isolating sample count while holding all other components fixed.

We distinguish a cell's *nominal* sample budget from the *effective* count used after a memory failure. In strict mode, successful runs contribute their metrics at the requested nominal cell, attempted failures follow the prespecified failure policy, and design exclusions remain absent; no result from a smaller

sample-budget cell is substituted. If fitting, or prediction after query batching, exhausts memory, the adaptive sensitivity mode searches smaller configured budgets for the same dataset, target, fold, model, and feature cap, approximately halving the count until a completed result is found.

Using only the budgeted training data, we remove features that are missing in the majority of rows. Missingness filtering may discard predictive information, and source-level curation does not guarantee nontriviality at every subsampled operating point. For each fold, we place the remaining columns in a seeded random order. Smaller feature caps are prefixes of this ordering, and we restore selected columns to registry order before fitting. Feature sets are thus nested across caps and identical across models within a fold and sample budget; each of the 5 folds uses a different ordering. Because missingness filtering is refitted after sample budgeting, the eligible feature set can differ across sample budgets. The grid measures the effect of retaining a larger random subset of the available representation. It is not a feature-selection benchmark.

Missing values are passed unchanged to models that handle them natively. For nearest neighbours, missing values are imputed with medians estimated from the training data. No transformation, allocation decision, or feature filter consults the held-out fold. Models that support only classification, or only regression, are evaluated on those tasks alone.

3.4 Cross-validation and leakage control

We use 5-fold cross-validation, with complementary folds shared across all cells and models. Sample caps restrict training data only. A single partition does not characterize sensitivity to alternative splits, particularly with few independent groups. When a loader supplies biological group identifiers, classification uses stratified group folds and regression uses group folds, so no group crosses the train-test boundary. If such identifiers are unavailable, classification uses stratified row-wise folds and regression uses row-wise folds. Without reliable group metadata, row-wise evaluation cannot establish patient- or site-level independence; duplicate screening does not exclude hidden biological dependence. The registry records the split unit explicitly; we do not infer it from apparent similarities among measurements.

3.5 Outcomes and aggregation

Macro-F1 is the primary classification metric, weighting classes equally without designating a positive class; RMSE is the primary regression metric. We save fold-level predictions and ground truth, then recompute the metrics during aggregation.

Within each cross-validation fold of each target, we compare every pair of eligible models using observed or failure-imputed scores. Because all comparisons are made within targets, classification and regression metrics are never placed on the same arithmetic scale. Under the Bradley-Terry model, the expected comparison score of model i against model j is

$$E_{ij} = \frac{1}{1 + 10^{(R_j - R_i)/400}}, \qquad R_i = 1000 + 400\,(\theta_i - \theta_{\mathrm{RF}}). \tag{1}$$

Wins, ties and losses are scored as 1, 0.5 and 0. Here θ_i is the latent ability of model i in base-10 log-odds units and R_i is its fitted Elo rating, therefore a difference of 400 rating points corresponds to 10:1 odds. Each fold comparison is weighted by the reciprocal of the target's number of folds, so every target contributes the same total weight regardless of endpoint type, while a model that wins three of five folds is distinguished from one that wins all five. For interpretability, we shift the ratings so that Random Forest has an Elo of 1,000. We resample the target pool 2,000 times and report the rounded median of those draws as the rating, with 95% percentile intervals from the same draws. We resample targets rather than folds, carrying each target's fold comparisons with it, because folds are repeated measurements of the same scientific task; the target-bootstrap intervals therefore do not fully capture split uncertainty. We score classification on macro-F1 because the shared adapter also uses it to score the internal validation split, so it governs validation-based model selection and early stopping wherever a model exposes them. Models fitted by their own criterion, including the linear, nearest-neighbour, and tree baselines, are scored on macro-F1 without being optimised for it, so the alignment between fitting objective and reported metric is not uniform across the roster. We also report mean macro-F1 and median fit time, the latter over successful fits alone, so cost is conditional on a completed fit while performance charges a failure at constant-predictor level; we

use Elo for cross-task ranking because it depends only on within-target comparisons (Bradley & Terry, 1952). Decision-threshold handling follows the AutoGluon adapter; Appendix E re-ranks the reference cell by AUROC. The anchor shift is applied inside every bootstrap draw, so each reported interval is an interval for that model's rating *relative to Random Forest*; the anchor's own interval is therefore degenerate. Because two such intervals share the anchor term, they are positively correlated and their overlap is not a test of one model against another. For a pairwise claim we instead take the difference between the two ratings within each draw, where the anchor cancels. A cell is designated complete when every scheduled unit in that cell has a terminal status, and the release manifest records the contributing cells and inputs. The reference-cell ranking in Figure 2 and Table 1 uses the strict nominal-cell export. The strict mode is also the default view on the interactive website, which additionally provides the adaptive sensitivity view and rankings under balanced accuracy, Matthews correlation, and AUROC. Both exports and the sample-fallback manifest are released.

4 RESULTS

The benchmark registry holds 43 datasets (30 classification and 13 regression), evaluated across 28 operating points. A dataset is included in a cell only if its training partition is at least as large as that cell's sample budget; otherwise, the fit would simply repeat the full-sample result. Consequently, the reference cell ranks 42 of them. TFMs generally occupy the leading ranks, and a paired bootstrap separates the leading model from its runner-up at the reference cell. Model order varies by operating point and biomedical modality, while fit time may favour a method other than the highest-rated one. We examine these patterns and their associated uncertainty below.

4.1 REFERENCE-CELL RANKING

The reference operating point is defined with p=10,000, n=100. We selected it before inspecting model results as it combines a high-dimensional feature budget with broad task coverage at a cohort size typical of the collection. At this point, the sample budget binds for 42 targets. The remaining dataset has fewer training rows than the cell's sample budget, so the cell would repeat their full-sample result and they do not enter the ranking. Figure 2 and Table 1 summarise the strict nominal-cell results.

RealTabPFN 2.5 has the highest point estimate, with Elo 1156 ([1098, 1212]). TabPFN 3 follows at 1070, a gap of 86 points. Resampling the target pool and taking the rating difference within each draw gives 87 Elo ([43, 129]), with RealTabPFN 2.5 rated above TabPFN 3 in all 2000 draws. The coarser target-level win count agrees. In their direct target-level comparison, RealTabPFN 2.5 wins 32 of 42 targets, ties two, and loses in eight. Counting a tie as one half, its empirical win score is 78.6% (target-bootstrap 95% interval [65.5%, 89.3%]); the interval excludes 50%. Mean classification performance orders the leading models differently: Logistic Regression reaches the highest macro-F1, 0.721, compared with 0.713 for RealTabPFN 2.5. The strongest TFM leads the rating, while a linear model remains a strong default in this biomedical HDLSS setting under macro-F1.

As runs that exhaust memory or exceed the one-hour time limit are imputed with constant predictor performance, the benchmark indirectly ranks models by deployability. It balances predictive accuracy with the ability to operate within a fixed resource budget, penalizing models that fail to fit a target instead of omitting them from the results

Failure counts differ substantially across models, so the two components should be read together; Appendix F reports recorded outcomes per model at this cell and Appendix F.1 re-ranks the cell with failures omitted instead of imputed.

4.2 MODEL ORDER DEPENDS ON THE OPERATING POINT AND MODALITY

Figure 3 compares six sample budgets at 10,000 features. From 20 to 1,000 training rows, the linear baseline falls from shared first place to 16th.

Feature width also changes model order. In a separate comparison on 34 shared targets at 100 training rows, increasing the feature budget from 2,000 to 10,000 moves TabICLv2 from rank 17 to 19 and Logistic Regression from 7 to 4. TabICLv2 and RealTabPFN 2.5 share a regular feature limit of 2,000,

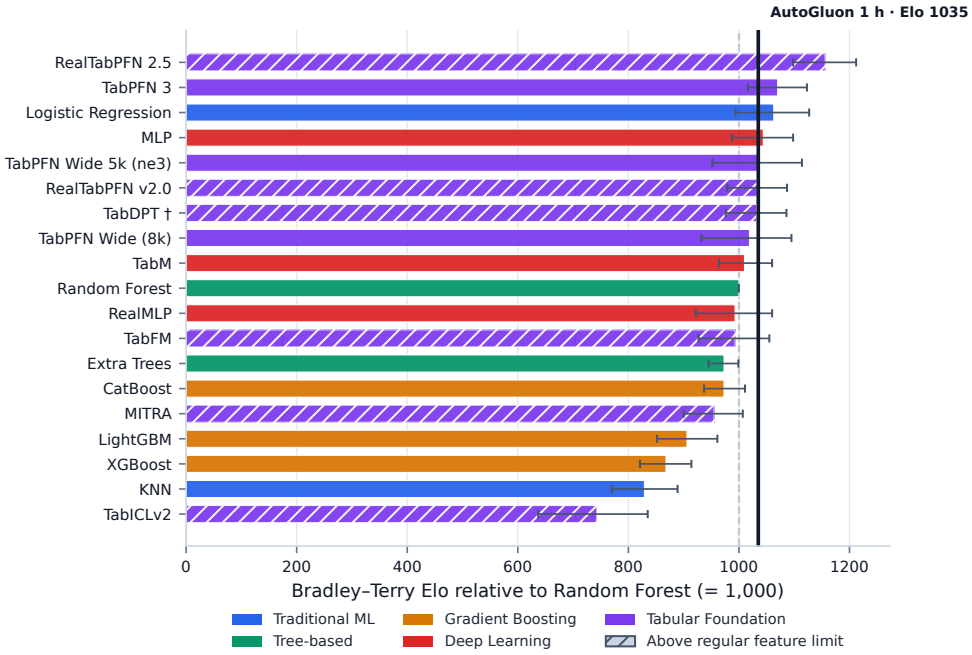

Figure 2: Strict reference-cell ranking. Bradley-Terry Elo at p=10,000, n=100, anchored at Random Forest = 1,000. Bars encode model families and error bars show target-bootstrap 95% intervals for each rating *relative to Random Forest*; the anchor is fixed at 1,000 in every draw, so its own interval has zero width. Overlap between two intervals is not a pairwise test; pairwise claims use the paired bootstrap difference. The solid vertical line gives the one-hour AutoGluon Extreme reference, evaluated on every eligible reference-cell target, and is not ranked as a peer. † TabDPT: Part of the benchmark training data was used in the training process of this model.

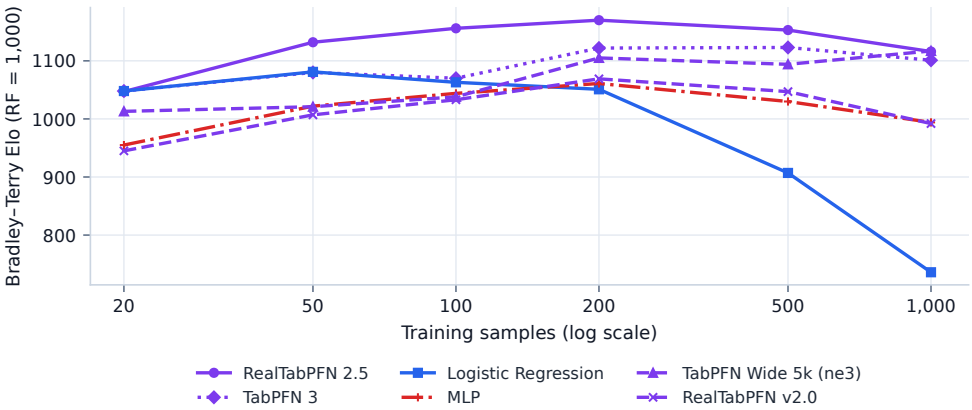

Figure 3: Sample-budget response at 10,000 features. Strict fold-level Elo using all eligible targets at each sample budget. Target coverage varies across budgets. The six leading peer models at the 100-sample reference cell are shown. Lines show bootstrap medians.

Rank	Model	Elo	95% CI	Targets	Macro-F1	RMSE ↓
1	RealTabPFN 2.5	1156	[1098, 1212]	42	0.713	1.743
2	TabPFN 3	1070	[1016, 1123]	42	0.685	1.837
3	Logistic Regression	1063	[994, 1127]	42	0.721	1.956
4	MLP	1044	[987, 1098]	42	0.701	2.116
5	TabPFN Wide 5k (ne3)	1038	[952, 1114]	30	0.675	–
6	RealTabPFN v2.0	1033	[979, 1087]	42	0.665	1.818
7	TabDPT	1032	[976, 1086]	42	0.694	2.095
8	TabPFN Wide (8k)	1019	[932, 1095]	30	0.668	–
9	TabM	1010	[964, 1060]	42	0.681	2.015
10	Random Forest	1000	[1000, 1000]	42	0.666	1.930
11	RealMLP	993	[922, 1060]	42	0.699	2.171
12	TabFM	993	[927, 1055]	42	0.633	1.791
13	Extra Trees	973	[945, 999]	42	0.644	1.998
14	CatBoost	973	[937, 1011]	42	0.645	1.978
15	MITRA	955	[900, 1007]	42	0.647	1.903
16	LightGBM	906	[852, 961]	42	0.639	1.881
17	XGBoost	868	[821, 914]	42	0.635	2.110
18	KNN	829	[770, 889]	42	0.598	2.220
19	TabICLv2	743	[637, 835]	42	0.360	2.168

Table 1: Strict reference-cell leaderboard. Macro-F1 is averaged over classification targets; raw RMSE over regression targets on their original scales. Elo combines within-target comparisons. Memory and time-limit failures are scored with the constant predictor; see Appendix F. Both TabPFN-Wide configurations are classification-only and cover 30 targets, not the 12 regression targets for which RMSE is measured.

yet RealTabPFN 2.5 remains first. Declared capacity alone does not predict the relative performance of these pipelines.

At the reference cell, RealTabPFN 2.5 leads gene expression, metagenomics, and molecular properties. TabFM leads the four embedding targets, and TabPFN Wide (8k) leads the methylation targets. For genomic prediction, Random Forest and Extra Trees lead the 4 regression targets. These pools contain at most 10 targets, so comparisons are descriptive.

Appendix B compares predictive performance and fit time.

5 Discussion

TFMs generally occupy the leading ranks, but the results do not support a single default model. Although the paired bootstrap separates RealTabPFN 2.5 from TabPFN 3 at the reference cell, the leading configuration varies across biomedical modalities. The strength of Logistic Regression here differs from the pattern reported by TabArena, where foundation models excel on smaller datasets and tuning and ensembling alter model-family comparisons. These two rankings are not directly comparable: TabArena covers a broader collection of general-purpose tables and includes tuned and ensembled variants, whereas we evaluate fixed configurations on biomedical HDLSS tasks. While the ranking differs from general-purpose benchmarks, the effects of domain, metric, tuning, and ensembling cannot be fully disentangled. Scoring also plays a role: under AUROC, the metric TabArena uses for binary classification, Logistic Regression drops out of the leading group and the top 6 places are held by tabular foundation models (Appendix E), so part of the contrast reflects threshold placement rather than the domain alone. One possible explanation is that pretraining provides a useful prior in small labelled cohorts, whereas regularisation keeps a linear decision rule competitive when the representation is already informative. The benchmark does not distinguish between these mechanisms.

The modality-specific comparisons show that a model should be matched to the representation and endpoint rather than taken from one overall ranking. The regression and embedding task pools remain smaller than the classification and omics task pools, so they warrant more caution.

6 LIMITATIONS

First, the overall ranking reflects the composition of this release, including its classification-regression balance and the uneven representation of biomedical modalities. The interactive leaderboard allows users to restrict comparisons to a modality of interest and inspect the corresponding performance and resource measurements.

Second, uniform feature capping is label-agnostic and reproducible but may omit sparse biological signals. The evaluation covers five seeded feature orderings rather than the full distribution; additional orderings would multiply the already large model-dataset-budget grid. Feature-width comparisons do not estimate the value of supervised feature selection.

Third, the comparison measures default behaviour rather than performance after equal hyperparameter tuning. AutoGluon's one-hour Extreme preset was evaluated only at the reference and full-feature, full-sample cells, as extending it across the grid would substantially increase compute. Neither comparison provides an upper bound for a model family. Sample-budget comparisons may also reflect changes in class allocation and training-dependent feature filtering, rather than sample count alone.

Fourth, target coverage decreases at larger fixed sample budgets because fewer datasets contain enough training observations. The full-sample condition instead uses all available training observations for each dataset and therefore restores coverage of smaller datasets. Its target pool differs from those of the fixed-budget conditions, so ranking differences cannot be attributed to training-set size alone. The paired rating difference is likewise model-based: Bradley-Terry assigns each model a single transitive ability and does not represent the modality- and cell-dependent ordering reported above, so its interval is conditional on that assumption. Additionally, rare-class filtering narrows the target definition and precludes conclusions about underrepresented classes.

Finally, three of the 43 benchmark datasets (OVA_Uterus, AP_Colon_Kidney, and AP_Ovary_Lung) originate from the GEMLeR/expO collection, which also supplied OVA_Lung and OVA_Endometrium used in TabDPT pretraining. TabDPT performance on these related datasets should therefore be interpreted with this overlap in mind. However, neither of the RealTabPFN models has any overlap.

7 CONCLUSION AND OUTLOOK

By evaluating tabular learners across heterogeneous biomedical measurements with many features and few samples, TabBench-Bio exposes behaviour that general-purpose suites rarely measure. Across the 43-dataset benchmark, TFMs generally occupy the leading ranks, and RealTabPFN 2.5 leads the reference-cell point ranking, followed by TabPFN 3 and Logistic Regression, whose point estimates are nearly identical. A paired bootstrap separates RealTabPFN 2.5 from TabPFN 3 at the reference cell, while modality-specific results show that the strongest configuration depends on the biological representation. Researchers should consider performance under conditions relevant to their data when selecting a model. TabBench-Bio supports this comparison and helps developers identify where their models perform reliably. As a living benchmark, it is currently expanding to include new models and tuned classical baselines. We welcome contributions of new biomedical tabular datasets, particularly from underrepresented assays and clinical endpoints, for inclusion in future releases.

REPRODUCIBILITY AND DATA AVAILABILITY

The anonymous interactive leaderboard and machine-readable exports are available at https://anonymous_iclr.codeberg.page/TabBench-Bio-Anonymous/. The locally generated embedding datasets are publicly available, including target labels, cross-validation group identifiers, and scripts documenting their construction. For each release, an aggregation manifest with cryptographic digests links the raw predictions and terminal status records to the generated tables, figures, and manuscript values. The evaluation protocol is described in Section 3, with further details on model defaults, failure rates, the dataset registry, and local embedding construction in Appendices A–G.1. Splits are frozen and reused across all models and operating points. Aggregation is deterministic, and the manifest described above links all recorded predictions and status records to the results in this manuscript.

ETHICS STATEMENT

This work evaluates published models on existing secondary datasets, including locally generated sequence-embedding datasets derived from existing biological sequences. No new human-subject data were collected and no institutional review board approval was required. No attempt at re-identification is made. Several sources derive from human cohorts, so the results describe default model behaviour on retrospective research data. The benchmark does not establish clinical validity, and no ranking reported here supports use in patient care.

AI USE STATEMENT

In this work, we used generative AI tools, specifically LLMs (mainly Claude Opus 5 and GPT-6 Astra) for coding and writing assistance. We have not used generative AI tools to propose or refine the research questions and hypotheses, to design the benchmark protocol, the operating-point grid, or the aggregation and rating methodology, to generate synthetic data, to formulate or prove mathematical claims, or to interpret the reported results; these were all carried out by the authors.

They were used to assist in implementing code by providing suggestions and generating code snippets, but not for full code generation. We used them for multiple code review steps. Additionally, we have reviewed all AI-assisted work. AI-assisted code was read by at least two authors and is covered by the repository's automated test suite. We take responsibility for the final content of this work, including all text, claims, and artefacts produced.

REFERENCES

Christina B. Azodi, Emily Bolger, Andrew McCarren, Mark Roantree, Gustavo de los Campos, and Shin-Han Shiu. Benchmarking parametric and machine learning models for genomic prediction of complex traits. *G3: Genes, Genomes, Genetics*, 9(11):3691–3702, 2019. doi: 10.1534/g3.119.400498.

Guy W. Bemis and Mark A. Murcko. The properties of known drugs. 1. molecular frameworks. *Journal of Medicinal Chemistry*, 39(15):2887–2893, 1996. doi: 10.1021/jm9602928.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. doi: 10.2307/2334029.

Garyk Brixi, Matthew G. Durrant, Jerome Ku, Mohsen Naghipourfar, Michael Poli, Gwanggyu Sun, Greg Brockman, Daniel Chang, Alison Fanton, Gabriel A. Gonzalez, et al. Genome modelling and design across all domains of life with Evo 2. *Nature*, 652:1349–1361, 2026. doi: 10.1038/s41586-026-10176-5.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016. doi: 10.1145/2939672.2939785.

Ron Edgar, Michael Domrachev, and Alex E. Lash. Gene expression omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*, 30(1):207–210, 2002. doi: 10.1093/nar/30.1.207.

Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. Autogluon-tabular: Robust and accurate automl for structured data. *arXiv preprint arXiv:2003.06505*, 2020.

Nick Erickson, Lennart Purucker, Andrej Tschalzev, David Holzmüller, Prateek Desai, David Salinas, and Frank Hutter. TabArena: A living benchmark for machine learning on tabular data. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-0519.

Gregory M. Findlay, Riza M. Daza, Beth Martin, Melissa D. Zhang, Anh P. Leith, Molly Gasperini, Joseph D. Janizek, Xingfan Huang, Lea M. Starita, and Jay Shendure. Accurate classification of BRCA1 variants with saturation genome editing. *Nature*, 562:217–222, 2018. doi: 10.1038/s41586-018-0461-z.

Anurag Garg, Muhammad Ali, Noah Hollmann, Lennart Purucker, Samuel Müller, and Frank Hutter. Real-tabpfn: Improving tabular foundation models via continued pre-training with real-world data. *arXiv preprint arXiv:2507.03971*, 2025.

Yury Gorishniy, Akim Kotelnikov, and Artem Babenko. TabM: Advancing tabular deep learning with parameter-efficient ensembling. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Sd4wYYOhmY.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In *Advances in Neural Information Processing Systems*, volume 35, pp. 507–520, 2022. doi: 10.52202/068431-0037.

Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Benjamin Jäger, Dominik Safaric, Simone Alessi, Adrian Hayler, Mihir Manium, Rosen Yu, Felix Jablonski, Shi Bin Hoo, Anurag Garg, Jake Robertson, Magnus Bühler, Vladyslav Moroshan, Lennart Purucker, Clara Cornu, Lilly Charlotte Wehrhahn, Alessandro Bonetto, Bernhard Schölkopf, Sauraj Gambhir, Noah Hollmann, and Frank Hutter. TabPFN-2.5: Advancing the state of the art in tabular foundation models. *arXiv preprint arXiv:2511.08667*, 2025. doi: 10.48550/arXiv.2511.08667. URL https://arxiv.org/abs/2511.08667.

Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Mihir Manium, Shi Bin Hoo, Magnus Bühler, Anurag Garg, et al. Tabpfn-3: Technical report. *arXiv preprint arXiv:2605.13986*, 2026.

Gregory Hannum, Justin Guinney, Ling Zhao, Li Zhang, Guy Hughes, Srini Sadda, Brandy Klotzle, Marina Bibikova, Jian-Bing Fan, Yuan Gao, Rob Deconde, Menzies Chen, Indika Rajapakse, Stephen Friend, Trey Ideker, and Kang Zhang. Genome-wide methylation profiles reveal quantitative views of human aging rates. *Molecular Cell*, 49(2):359–367, 2013. doi: 10.1016/j.molcel.2012.10.016.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637:319–326, 2025. doi: 10.1038/s41586-024-08328-6.

David Holzmüller, Léo Grinsztajn, and Ingo Steinwart. Better by default: Strong pre-tuned MLPs and boosted trees on tabular data. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-0837.

Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W. Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Artificial intelligence foundation for therapeutic science. *Nature Chemical Biology*, 18(10):1033–1036, 2022. doi: 10.1038/s41589-022-01131-2.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30, pp. 3146–3154, 2017.

Mario Koddenbrock, Christoph Lange, Robin Legner, Martin Jäger, Martin Kögler, Mariano N. Cruz Bournazou, Peter Neubauer, Felix Biessmann, and Erik Rodner. RamanBench: A large-scale benchmark for machine learning on raman spectroscopy. *arXiv preprint arXiv:2605.02003*, 2026. doi: 10.48550/arXiv.2605.02003. URL https://arxiv.org/abs/2605.02003.

Christopher Kolberg, Jules Kreuer, Jonas Huurdeman, Sofiane Ouaari, Katharina Eggensperger, and Nico Pfeifer. TabPFN-Wide: Continued pre-training for extreme feature counts. *arXiv preprint arXiv:2510.06162*, 2025. doi: 10.48550/arXiv.2510.06162. URL https://arxiv.org/abs/2510.06162.

Weihao Kong and Abhimanyu Das. Introducing TabFM: A zero-shot foundation model for tabular data. Google Research Blog, 2026. URL https://research.google/blog/introducing-tabfm-a-zero-shot-foundation-model-for-tabular-data/. Accessed 25 August 2026.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023. doi: 10.1126/science.ade2574.

Si-Yang Liu, Hao-Run Cai, Qi-Le Zhou, Huai-Hong Yin, Tao Zhou, Jun-Peng Jiang, and Han-Jia Ye. TALENT: A tabular analytics and learning toolbox. *Journal of Machine Learning Research*, 26 (226):1–16, 2025. URL https://www.jmlr.org/papers/v26/25-0512.html.

Yun Liu, Martin J. Aryee, Leonid Padyukov, M. Daniele Fallin, Espen Hesselberg, Arni Runarsson, Lovisa Reinius, Nathalie Acevedo, Margaret Taub, Marcus Ronninger, Klementy Shchetynsky, Annika Scheynius, Juha Kere, Lars Alfredsson, Lars Klareskog, Tomas J. Ekström, and Andrew P. Feinberg. Epigenome-wide association data implicate DNA methylation as an intermediary of genetic risk in rheumatoid arthritis. *Nature Biotechnology*, 31(2):142–147, 2013. doi: 10.1038/nbt.2487.

Junwei Ma, Valentin Thomas, Rasa Hosseinzadeh, Alex Labach, Hamidreza Kamkari, Jesse C. Cresswell, Keyvan Golestan, Guangwei Yu, Anthony L. Caterini, and Maksims Volkovs. TabDPT: Scaling tabular foundation models on real data. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5748.

Alex L. Mitchell, Alexandre Almeida, Martin Beracochea, Miguel Boland, Joseph Burgin, Guy Cochrane, Michael R. Crusoe, Varsha Kale, Simon C. Potter, Lorna J. Richardson, Ekaterina Sakharova, Maxim Scheremetjew, Anton Korobeynikov, Alex Shlemov, Olga Kunyavskaya, Alla Lapidus, and Robert D. Finn. MGnify: The microbiome analysis resource in 2020. *Nucleic Acids Research*, 48(D1):D570–D578, 2020. doi: 10.1093/nar/gkz1035.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do bayesian inference. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=KSugKcbNf9.

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. CatBoost: Unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems*, volume 31, pp. 6638–6648, 2018.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A tabular foundation model for in-context learning on large data. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 50817–50847. PMLR, 2025. URL https://proceedings.mlr.press/v267/qu25d.html.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICLv2: A better, faster, scalable, and open tabular foundation model. *arXiv preprint arXiv:2602.11139*, 2026. doi: 10.48550/arXiv.2602.11139. URL https://arxiv.org/abs/2602.11139.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010. doi: 10.1021/ci100050t.

Shriyank Somvanshi, Subasish Das, Syed Javed, Gian Antariksa, and Ahmed Hossain. A survey on tabular data: from tree-based methods to tabular deep learning. *ACM Computing Surveys*, 2026. doi: 10.1145/3807777. URL https://dl.acm.org/doi/abs/10.1145/3807777.

Vineet Thumuluri, José Juan Almagro Armenteros, Alexander Rosenberg Johansen, Henrik Nielsen, and Ole Winther. DeepLoc 2.0: Multi-label subcellular localization prediction using protein language models. *Nucleic Acids Research*, 50(W1):W228–W234, 2022. doi: 10.1093/nar/gkac278.

Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luis Torgo. OpenML: Networked science in machine learning. *ACM SIGKDD Explorations Newsletter*, 15(2):49–60, 2014. doi: 10.1145/2641190.2641198.

John N Weinstein, Eric A Collisson, Gordon B Mills, Kenna R Mills Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, and Joshua M Stuart. The cancer genome atlas pan-cancer analysis project. *Nature Genetics*, 45(10):1113–1120, 2013. doi: 10.1038/ng.2764.

Minghao Xu, Zuobai Zhang, Jiarui Lu, Zhaocheng Zhu, Yangtian Zhang, Chang Ma, Runcheng Liu, and Jian Tang. PEER: A comprehensive and multi-task benchmark for protein sequence understanding. In *Advances in Neural Information Processing Systems*, volume 35, pp. 35156–35173, 2022. doi: 10.52202/068431-2548.

Barbara Zdrazil, Eloy Felix, Fiona Hunter, Emma J. Manners, James Blackshaw, Sybilla Corbett, Marleen de Veij, Harris Ioannidis, David Mendez Lopez, Juan F. Mosquera, Maria Paula Magariños, Nicolas Bosc, Ricardo Arcila, Tevfik Kizilören, Anna Gaulton, A. Patrícia Bento, Melissa F. Adasme, Peter Monecke, Gregory A. Landrum, and Andrew R. Leach. The ChEMBL database in 2023: A drug discovery platform spanning multiple bioactivity data types and time periods. *Nucleic Acids Research*, 52(D1):D1180–D1192, 2024. doi: 10.1093/nar/gkad1004.

Xiyuan Zhang, Danielle C. Maddix, Junming Yin, Nick Erickson, Abdul Fatir Ansari, Boran Han, Shuai Zhang, Leman Akoglu, Christos Faloutsos, Michael W. Mahoney, Cuixiong Hu, Huzefa Rangwala, George Karypis, and Bernie Wang. Mitra: Mixed synthetic priors for enhancing tabular foundation models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-0535.

A MODEL CONFIGURATION DEFAULTS

Table 2 lists the key defaults of the benchmark model configurations. Maximum iteration counts are subject to model-native early stopping and the common one-hour timeout. Peer models use no hyperparameter optimisation or AutoGluon bagging; AutoGluon Extreme is an external reference. Parameters not listed remain at the corresponding library defaults.

Model	Important default parameters
Constant	Class-prior predictor (classification); mean predictor (regression)
CatBoost	Learning rate 0.05; at most $10{,}000$ iterations; adaptive early stopping
LightGBM	Learning rate 0.05; at most $10{,}000$ rounds; adaptive early stopping
KNN	$k = 5$; uniform weights; Euclidean distance
Logistic Regression	L_2 penalty; $C = 1$ (classification); $\alpha = 1$ (regression); standardised inputs
Random Forest	300 trees; bootstrap sampling; at most $15{,}000$ leaf nodes
XGBoost	Tree booster; learning rate 0.1; at most $10{,}000$ rounds; adaptive early stopping
Extra Trees	300 trees; bootstrap sampling; at most $15{,}000$ leaf nodes
MLP	Four layers of width 128; ReLU; dropout 0.1; Adam; learning rate 3×10^{-4}; at most $1{,}000$ epochs
RealMLP	TD preset; three layers of width 256; 256 epochs; learning rate 0.04 (classification) or 0.2 (regression)
TabM	TabM-mini; 32 subnetworks; block width 512; dropout 0.1; learning rate 2×10^{-3}; patience 16
MITRA	Tab2D; one estimator; 50 fine-tuning steps; learning rate 10^{-4}; patience 40
RealTabPFN v2.0	Default v2 checkpoint; 8 estimators; 2 features per group; temperature 0.9
RealTabPFN 2.5	Default v2.5 checkpoint; 8 estimators; 3 features per group; temperature 0.9
TabPFN 3	Default v3 checkpoint; 8 estimators; feature-group size 3; temperature 0.9
TabPFN Wide (8k)	`wide-v2-8k` checkpoint; one estimator; one feature per group
TabPFN Wide 5k (ne3)	`wide-v2-5k` checkpoint; 3 estimators; one feature per group
TabFM	Default v1.0 PyTorch checkpoint; 32 estimators; random feature permutations; no feature cap; averaged logits
TabDPT	Default v1.2 checkpoint; 8 ensembles; full training context; standard normalisation; PCA feature reduction; temperature 1
TabICLv2	TabICLv2 checkpoint; 8 estimators; Latin feature shuffling; shifted classes; averaged logits
AutoGluon	Extreme preset; one-hour budget; native model portfolio and weighted ensemble

Table 2: Principal default parameters of the benchmark model configurations. Parameters not listed remain at the corresponding library defaults.

B PERFORMANCE AND FIT TIME

At the reference cell, RealTabPFN 2.5 has a median fit time of 78.9 seconds per fold. Logistic Regression at second place with just 33.1 seconds. Times cover successful fits and include the shared harness overhead. They exclude prediction, pretraining, memory, and energy costs. The dashed frontier joins methods for which no other plotted method is both at least as accurate and faster to fit.

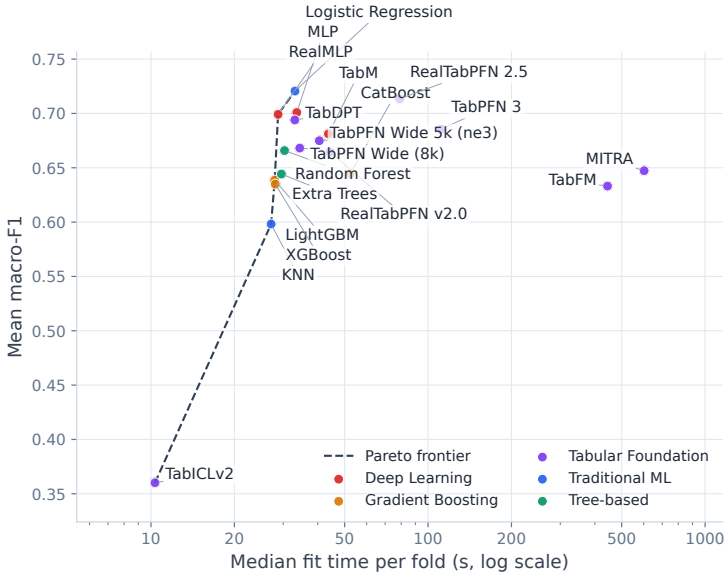

Figure A1: Mean macro-F1 against median fit time at p=10,000, n=100. The time axis is logarithmic.

C FULL-FEATURE, FULL-SAMPLE RANKING

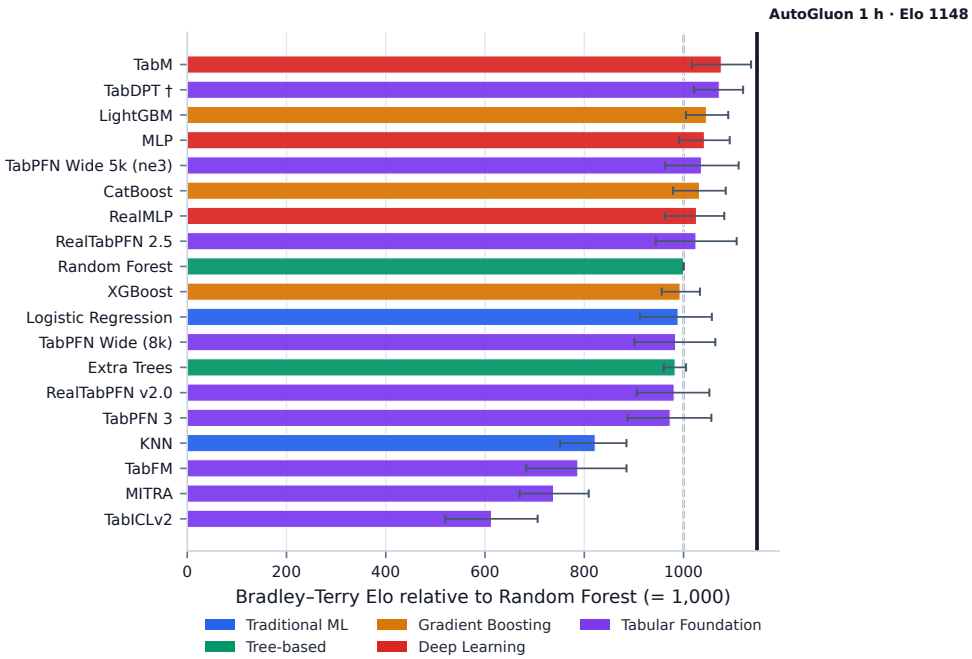

Figure A2: Strict fold-level Elo using all available training rows and features after shared preprocessing. Most models cover 43 targets; the classification-only TabPFN-Wide models cover 30. Error bars show 95% target-bootstrap intervals relative to Random Forest (= 1,000). The solid vertical line marks the one-hour AutoGluon Extreme reference. † TabDPT: Part of the benchmark training data was used in the training process of this model.

D LEADING MODELS ACROSS THE GRID

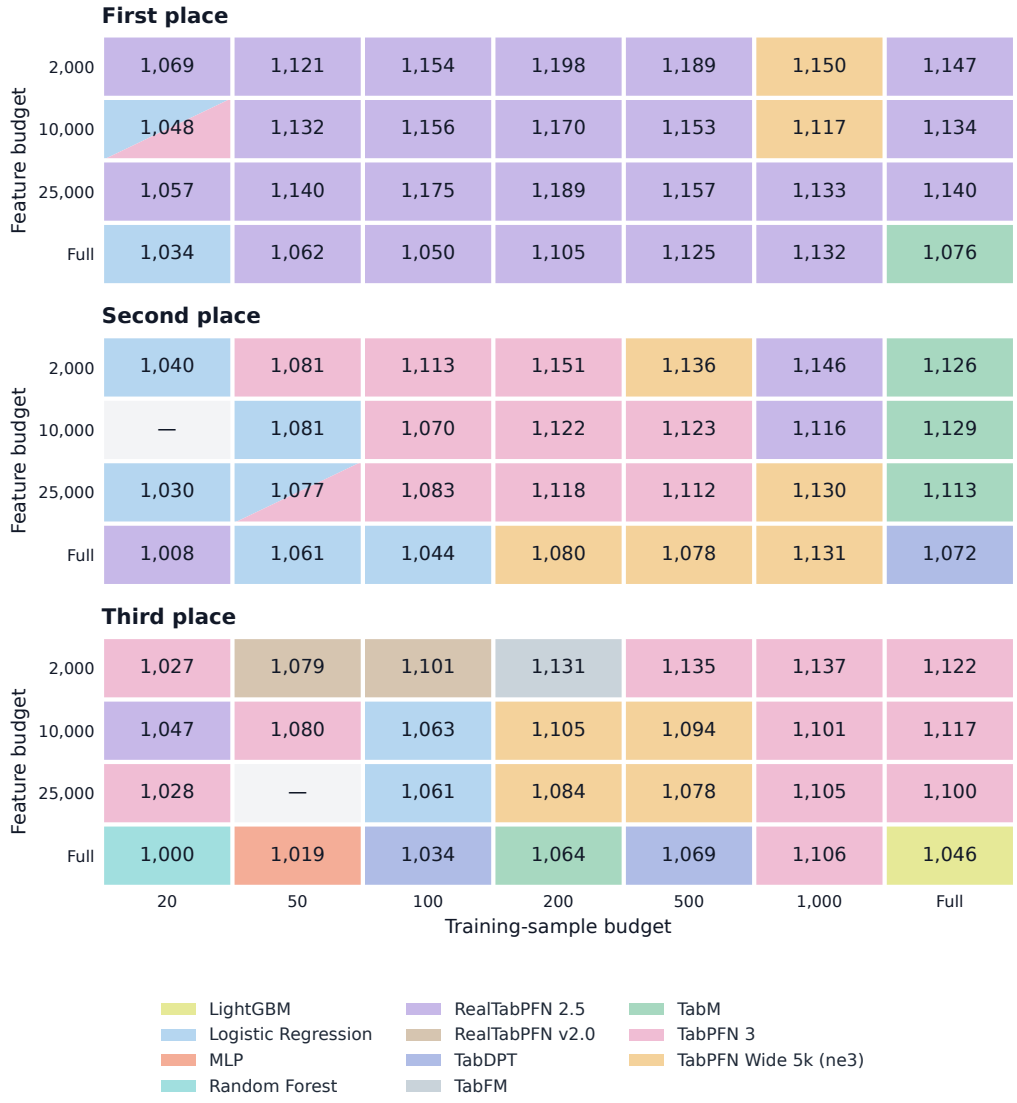

Figure A3: First, second and third place at each grid point under strict fold-level Elo. Colours identify models; numbers give rounded bootstrap-median Elo. Split cells denote ties; subsequent ranks are skipped and marked by a dash. AutoGluon is excluded as an external reference. Target coverage varies across cells; rank differences do not establish statistical significance. Full uses all available training rows or features after shared preprocessing.

E METRIC SENSITIVITY

Macro-F1 combines discrimination with threshold placement; decision-threshold handling follows the AutoGluon adapter. Figure A4 re-orders the reference cell by AUROC as a sensitivity check, keeping regression on RMSE. AUROC is also the metric TabArena uses for binary classification, so the figure shows how much of the contrast with that benchmark depends on scoring.

The ranking changes. RealTabPFN 2.5 has the highest AUROC point estimate, ahead of TabPFN Wide 5k (ne3) and TabPFN 3, with the three separated by 20 Elo. The larger change is below them: Logistic Regression, third on macro-F1, falls behind the leading 6 places, all of which are held by tabular foundation models. Comparisons between model families must therefore be read together with the metric under which they are computed.

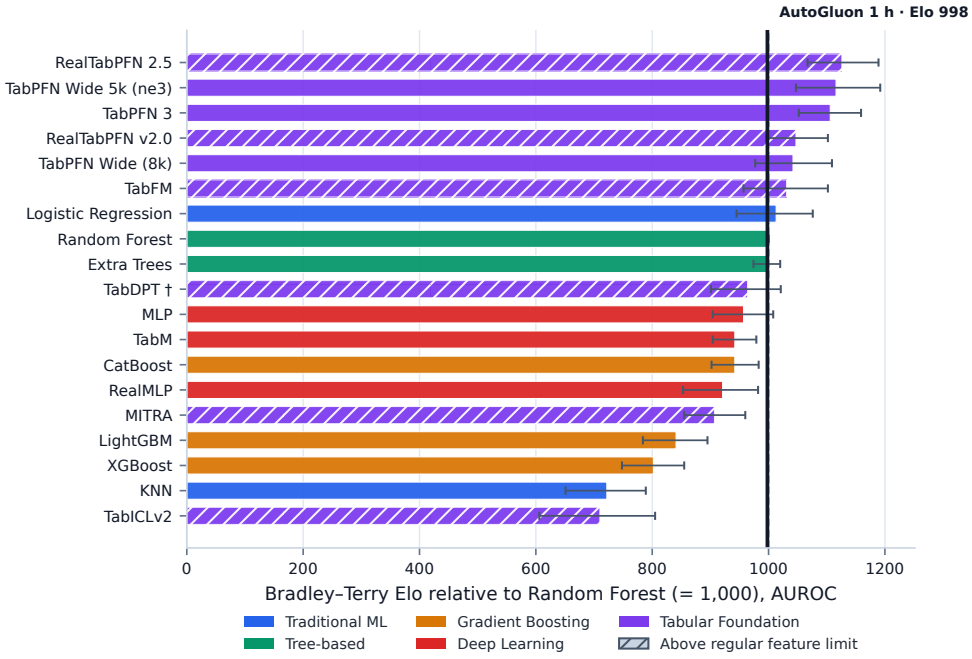

Figure A4: Reference-cell ranking ordered by AUROC. Identical runs, folds, and estimator as Figure 2; only the classification score changes, recomputed from the stored fold-level predictions without refitting. AUROC is applied to every classification target, one-vs-rest for the multiclass ones. This is a post-hoc metric sensitivity analysis using fits selected or early-stopped based on macro-F1; AUROC-based selection could change the results, and unchanged predictions do not imply unchanged rankings. AutoGluon Extreme additionally selects its ensemble weights on macro-F1 and is reported as a reference rather than a ranked peer.

F RECORDED FAILURE RATES

Table 3 reports the recorded terminal outcomes per model at the reference cell. Memory and time-limit failures are scored with the constant predictor and therefore enter the rankings in Figure 2 and Table 1; design skips do not. Failure counts are concentrated in a few models rather than spread across the roster, so the deployability component of the ranking affects some models far more than others.

Model	Pass/attempt	Fail (%)	Memory	Timeout	Other	Skip
Constant	210/210	0.0	0	0	0	5
CatBoost	203/210	3.3	0	7	0	5
LightGBM	210/210	0.0	0	0	0	5
KNN	160/210	23.8	0	0	50	5
Logistic Regression	210/210	0.0	0	0	0	5
Random Forest	210/210	0.0	0	0	0	5
XGBoost	210/210	0.0	0	0	0	5
Extra Trees	210/210	0.0	0	0	0	5
MLP	210/210	0.0	0	0	0	5
RealMLP	210/210	0.0	0	0	0	5
TabM	210/210	0.0	0	0	0	5
MITRA	205/210	2.4	0	0	5	5
RealTabPFN v2.0	205/210	2.4	0	0	5	5
RealTabPFN 2.5	205/210	2.4	0	0	5	5
TabPFN 3	205/210	2.4	0	0	5	5
TabPFN Wide (8k)	145/150	3.3	0	0	5	65
TabPFN Wide 5k (ne3)	145/150	3.3	0	0	5	65
TabFM	205/210	2.4	0	0	5	5
TabDPT	210/210	0.0	0	0	0	5
TabICLv2	55/210	73.8	155	0	0	5
AutoGluon (ref.)	210/210	0.0	0	0	0	5

Table 3: Recorded failure rates at p=10,000, n=100. Counts are recorded terminal dataset–fold outcomes at the nominal cell, before adaptive fallback. Failure percentage is $100F/(P+F)$; design skips are excluded from its denominator. Memory denotes recorded `fit_oom` or `inference_oom`, including possible estimator refusals; timeout denotes `time_limit`. Other includes generic errors and unspecified reasons; recorded codes do not establish root causes. Category and skip columns are counts. A dash denotes no attempts or no recorded outcomes, not zero failure. Missing runs are not counted; this table does not certify completion. Models may have different eligible task pools. AutoGluon is a separate reference.

F.1 FAILURE-POLICY SENSITIVITY

The primary ranking scores failures at the constant predictor. Figure A5 instead omits failures and retains only model–target pairs with all scheduled folds scored. Models therefore cover different target pools. This sensitivity analysis does not establish bounds on ratings or ranks.

The rank correlation between the two policies is 0.72, and 16 models change rank. TabICLv2 moves the most: 17 places after losing 31 targets. RealTabPFN 2.5 still leads, followed by TabICLv2.

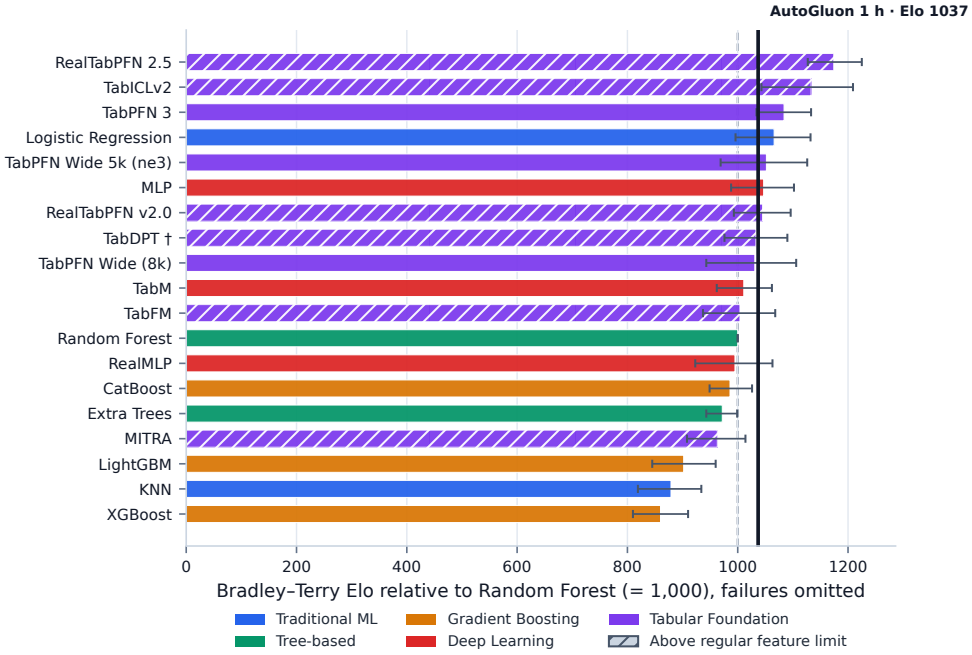

Figure A5: Reference-cell ranking with failures omitted. Runs, folds, metric and estimator are unchanged from Figure 2; no models are refitted. Omitting failures removes affected model–target pairs, so target coverage differs between models. † TabDPT: Part of the benchmark training data was used in the training process of this model.

G DATASET REGISTRY

Table 4 is generated from the dataset configuration and versioned registry used for the benchmark. Registry IDs remain stable even when a source title is abbreviated; the generated metadata CSV also retains source and licence fields. Samples counts the rows the benchmark evaluates, after classes with fewer than ten samples are removed. Features is the native width of the source table, before any feature cap is applied, so a dataset wider than a cell's budget contributes only a random subset of these columns in that cell.

Registry ID	Dataset	Target	Modality	Task	Samples	Features
OpenML-1138	Uterus vs. Other Tissue	Tissue	Expr.	C	1,545	10,935 ↗
OpenML-1458	ARCENE Serum Cancer	Class	Other	C	200	10,000 ↗
TCGA-TCGA-BRCA_Gene-Expression-Quantification	TCGA Breast (BRCA, RNA-seq)	sample_type	Expr.	C	1,231	60,660 ↗
GEO-GSE10893	Breast Cancer Subtypes (GSE10893)	subtype	Expr.	C	165	109,315 ↗
TCGA-TCGA-HNSC_Gene-Expression-Quantification	TCGA Head & Neck (HNSC, RNA-seq)	sample_type	Expr.	C	566	60,660 ↗
local-BRCA_Evo2-VEP	BRCA VEP (Evo2)	y	Embed.	C	3,893	8,192 ↗
local-DeepLoc2-Fungi-ESM2	DeepLoc2 Fungi (ESM-2)	membrane	Embed.	C	5,841	1,280 ↗
OpenML-1083	mouseType	Decision	Expr.	C	214	45,101 ↗
OpenML-1106	GCM	class	Expr.	C	190	16,063 ↗
OpenML-1137	AP_Colon_Kidney	Tissue	Expr.	C	546	10,935 ↗
OpenML-1084	BurkittLymphoma	Decision	Expr.	C	220	22,283 ↗
OpenML-1088	variousCancers_final	Decision	Expr.	C	383	54,675 ↗
OpenML-1140	AP_Ovary_Lung	Tissue	Expr.	C	324	10,935 ↗
MGYS00005384	MGnify functional profile	host sex	Metag.	C	674	17,026 ↗
MGYS00005380	MGnify functional profile	source	Metag.	C	264	16,690 ↗
MGYS00005285	MGnify functional profile	host sex	Metag.	C	438	12,705 ↗
MGYS00006282	MGnify functional profile	host sex	Metag.	C	256	12,968 ↗
MGYS00002425	MGnify functional profile	host sex	Metag.	C	394	12,651 ↗
MGYS00005381	MGnify functional profile	environment (biome)	Metag.	C	218	13,394 ↗
MGYS00001255	MGnify functional profile	host sex	Metag.	C	152	14,103 ↗
MGYS00001056	MGnify functional profile	source	Metag.	C	306	15,268 ↗
MGYS00005379	MGnify functional profile	environment (material)	Metag.	C	1,450	14,826 ↗
gut-cirrhosis	Gut microbiome markers (healthy vs cirrhosis)	disease	Metag.	C	232	33,539 ↗
TDC-BBB-Martins	Blood-Brain Barrier Penetration (Martins)	Y	Mol.	C	1,954	2,048 ↗
TDC-AMES	Ames Mutagenicity	Y	Mol.	C	7,255	2,048 ↗
TDC-CYP3A4-Veith	CYP3A4 Inhibition (Veith)	Y	Mol.	C	12,300	2,048 ↗
GEO-GSE42861-Methylation-RA	Rheumatoid Arthritis DNA Methylation (GSE42861)	disease state	Meth.	C	689	30,000 ↗
FusionAI-NTv2-GeneFusion	Gene Fusion Breakpoints (NT-v2)	fusion	Embed.	C	51,859	2,048 ↗
GEO-GSE50660-Methylation-Smoking	Smoking Status DNA Methylation (GSE50660)	smoking (0, 1 and 2, which represent never, former and current smokers)	Meth.	C	464	30,000 ↗
GEO-GSE147221-Methylation-Schizophrenia	Schizophrenia DNA Methylation (GSE147221)	status	Meth.	C	679	30,000 ↗
OpenML-46983	Riboflavin Production	x	Expr.	R	71	4,088 ↗

Registry ID	Dataset	Target	Modality	Task	Samples	Features
OpenML-430	mtp2	oz1143	Other	R	274	1,142 ↗
TDC-AqSolDB	Aqueous Solubility (AqSolDB)	Y	Mol.	R	9,562	2,048 ↗
TDC-Lipophilicity-AstraZeneca	Lipophilicity (AstraZeneca)	Y	Mol.	R	4,200	2,048 ↗
TDC-LD50-Zhu	Acute Toxicity LD50 (Zhu)	Y	Mol.	R	7,341	2,048 ↗
GEO-GSE40279-Methylation-Age	Blood DNA Methylation Age (GSE40279)	age (y)	Meth.	R	656	30,000 ↗
local-PEER-BetaLactamase-ESM2	TEM-1 Activity (ESM-2)	activity	Embed.	R	5,198	1,280 ↗
gp-maize-FT	Maize Flowering Time Genomic Prediction	FT	SNP	R	391	244,781 ↗
gp-rice-FT	Rice Flowering Time Genomic Prediction	FT	SNP	R	327	57,542 ↗
gp-soy-YLD	Soybean Yield Genomic Prediction	YLD	SNP	R	5,014	4,234 ↗
gp-switchgrass-HT	Switchgrass Height Genomic Prediction	HT	SNP	R	514	217,150 ↗
CHEMBL206	Oestrogen receptor bioactivity	pchembl_value	Mol.	R	4,544	2,048 ↗
CHEMBL2034	Glucocorticoid Receptor Bioactivity	pchembl_value	Mol.	R	3,127	2,048 ↗

Table 4: Dataset registry. C: classification; R: regression. Arrows link to dataset sources; for locally constructed embeddings, they link to Appendix G.1.

G.1 CONSTRUCTION OF LOCAL EMBEDDING DATASETS

We constructed three sequence-representation datasets and loaded the resulting numerical tables through the same registry interface as the remotely retrieved datasets. The target and biological group columns were removed from the feature matrix and retained separately for evaluation and grouped cross-validation.

For the BRCA1 variant-effect task, we used the functional measurements of Findlay et al. (2018), also used in the Evo 2 BRCA1 evaluation (Brixi et al., 2026). Our prepared input CSV contains paired reference and variant DNA sequences and the three-level functional label (FUNC, LOF, or INT). Each sequence was centre-cropped to the 8,192-base-pair context used in this benchmark and embedded with Evo2-7B at layer `blocks.26.mlp.l3`. We mean-pooled the token-level representation of each sequence and concatenated its 4,096-dimensional reference and variant vectors to form an 8,192-dimensional feature vector. A truncated SHA-256 hash of the reference sequence identified variants from the same genomic context and served as the cross-validation group.

For the protein tasks, source files were downloaded from their official hosts and verified against pinned checksums. We embedded proteins with the pinned `facebook/esm2_t33_650M_UR50D` revision (Lin et al., 2023) and mean-pooled its 1,280-dimensional residue representations. Proteins longer than the model context were split into non-overlapping 1,022-residue chunks and pooled using residue-count weighting, so every residue contributed once. The DeepLoc 2.0 dataset contains all 5,841 unique fungal proteins in the official Swiss-Prot training and validation table; membrane association is the binary target, and the five published homology partitions define the cross-validation groups (Thumuluri et al., 2022). The PEER TEM-1 dataset combines the official training, validation, and test files into 5,198 unique single-mutant sequences; the scaled experimental activity is the regression target, and the position of the sole amino-acid substitution defines 286 cross-validation groups (Xu et al., 2022). The generated protein tables were validated for finite, non-constant features and accompanied by manifests that record the model revision, source and output digests, pooling procedure, and dimensions.