AUTOTABBENCH: AUTOMATED TABULAR BENCHMARK FOR TABULAR FOUNDATION MODELS WITH EVIDENCE-GROUNDED SCOPE ANALYSIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Tabular foundation models (TFMs), pretrained on large-scale tabular data, have emerged as powerful data-analysis assistants for real-world applications, supporting data interpretation, prediction, and decision-making across diverse domains. As both general-purpose and domain-specific TFMs continue to grow, a natural question arises: how can we comprehensively and systematically evaluate their capabilities and determine which models are best suited to different tasks and conditions? Existing benchmarks provide the main basis for such comparisons, but they often reduce model evaluation to aggregate predictive performance over limited conditions. As a result, they provide limited evidence about how model advantages change across task types, distribution shifts, data characteristics, and perturbations. We argue that reliable evaluation requires making explicit the boundary of conditions under which a benchmark supports its conclusions.

To this end, we introduce **BenchScope**, an evidence-grounded framework for benchmark scope analysis and design. BenchScope recovers verifiable evaluation evidence from papers, code, configurations, and manifests, and structures it across task, generalization, and protocol dimensions. Using explicit evidence requirements and decision rules, it characterizes benchmark boundaries by identifying covered conditions, missing coverage, protocol constraints, and unresolved evidence, and translates these findings into benchmark-construction requirements. Across six benchmarks, structured boundary recovery improves from 80.56% to 94.44%. Applying BenchScope to 37 benchmark entries, we construct **AutoTabBench**, an 80-task automated tabular benchmark covering routine prediction, natural distribution shifts, and controlled stress conditions, under which model advantages vary substantially across evaluation settings.

1 INTRODUCTION

Tabular foundation models (TFMs) increasingly challenge task-specific learners on predictive benchmarks (Hollmann et al., 2025; Qu et al., 2025; QU et al., 2026; Zhang et al., 2025), renewing interest in comparisons across datasets and model families (Grinsztajn et al., 2022; McElfresh et al., 2023; Liu et al., 2025; Erickson et al., 2026). The conclusions depend on the evaluations behind these comparisons: task selection affects relative performance, while temporal and domain shifts test generalization beyond routine prediction (Dehghani et al., 2021; Rubachev et al., 2025; Gardner et al., 2023; Purucker et al., 2026). This raises a benchmark-level question: which evaluation conditions support a reported model advantage, and which tests assess its reach?

A dataset name alone leaves the prediction target, split, preprocessing, and evaluation-time information unspecified. Inventories and documentation provide essential context (Bischl et al., 2021; Gebru et al., 2021; Vanschoren et al., 2014), but these task-level conditions determine whether two evaluations repeat the same evidence or test complementary settings. We call the scope supported by a benchmark's data, tasks, and protocol its *benchmark boundary*. Recovering this boundary turns scattered documentation into criteria for interpreting comparisons and choosing tests.

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.

We introduce *BenchScope*,[1] an evidence-grounded framework for recovering and characterizing benchmark scope. BenchScope combines agent-assisted evidence recovery, script-based verification of factual quantities, and a reusable boundary analysis that specifies evidence requirements and decision rules. Together, these components reconcile papers, code, configurations, and manifests into structured profiles of task coverage, generalization conditions, evaluation protocols, and evidence states. We use these profiles to automate redundancy checks, complementary task selection, and protocol design, assembling *AutoTabBench* with 58 routine-prediction tasks, 22 natural-shift tasks, and eight controlled stress families covering class imbalance, feature masking, missingness, numerical perturbations, unseen categories, label noise, and training-sample reduction.

Our contributions are fourfold:

- **Problem setup.** We formulate benchmark scope as the task, generalization, protocol, and evidence conditions under which benchmark conclusions hold, and show across 37 audited entries that TFM evaluations differ in consequential conditions that are left implicit.
- **BenchScope.** We introduce BenchScope, which recovers conditions from papers, code, configurations, and manifests into evidence-linked boundary profiles. Boundary Skill raises structured recovery accuracy from 80.56% to 94.44% across six benchmarks.
- **AutoTabBench.** We assemble AutoTabBench from recovered scope profiles, yielding 58 routine-prediction tasks, 22 natural-shift tasks, and eight controlled stress families selected and implemented without using model scores.
- **Insight.** We show that routine benchmarks are not shift benchmarks because they omit distribution changes; routine advantage does not imply robustness because gains can vanish under perturbations; and aggregation/reporting scope can change model conclusions because it reweights tasks, failures, and evaluation conditions.

2 Related Work

Tabular foundation models and their evaluation. TabPFN, TabICL/TabICLv2, and LimiX develop pretrained predictors that transfer across tabular tasks (Hollmann et al., 2025; Qu et al., 2025; QU et al., 2026; Zhang et al., 2025); TP-BERTa, TabuLa, and TabSTAR additionally exploit language representations or textual fields (Yan et al., 2024; Gardner et al., 2024; Arazi et al., 2025). Their evaluation builds on comparisons of neural and tree-based learners and reusable model-evaluation repositories (Gorishniy et al., 2021; Grinsztajn et al., 2022; McElfresh et al., 2023; Salinas & Erickson, 2024). OpenML suites, PMLB, AMLB, TALENT, and TabArena organize datasets, implementations, and evaluation protocols at increasing scale (Bischl et al., 2021; Fischer et al., 2023; Romano et al., 2022; Gijsbers et al., 2024; Liu et al., 2025; Erickson et al., 2026). Alongside predictive accuracy, recent studies examine conditional density estimation and uncertainty reliability (Izbicki & Rodrigues, 2026; De Melo Costa et al., 2026).

Coverage and generalization conditions. Specialized evaluations isolate conditions that broad predictive leaderboards can aggregate together: scarce data in PMLBmini, temporal change in TabReD, domain shifts in TableShift, and feature shifts in TabFSBench (Knauer et al., 2024; Rubachev et al., 2025; Gardner et al., 2023; Cheng et al., 2025). BeyondArena and recent TFM OOD evaluations directly examine generalization across settings (Purucker et al., 2026; Loza et al., 2026), complementing the application-grounded distribution shifts organized by WILDS (Koh et al., 2021). Text- and image-enriched tabular benchmarks broaden the input conditions under study (Shi et al., 2021; Tang et al., 2024; Arazi et al., 2026). These evaluations cover complementary settings.

Documentation and automated benchmark auditing. Datasheets, OpenML, and HELM structure dataset documentation, experiment records, and scenario–metric coverage, respectively (Gebru et al., 2021; Vanschoren et al., 2014; Liang et al., 2023). Agent-based auditing examines benchmark artifacts for inconsistencies in task specifications, execution environments, and evaluation logic (Wang et al., 2026; Tu et al., 2026). For tabular evaluation, target, split, and preprocessing jointly determine task identity. BenchScope recovers these conditions into comparable boundary profiles, distinguishing covered conditions, protocol judgments, and unresolved evidence.

[1] BenchScope and AutoTabBench will be released upon acceptance.

Agent-driven construction and evaluation-set selection. Agent-driven construction combines targeted data generation, verification, and evaluation across general benchmarks and software requirements (Li et al., 2025; Butt et al., 2025; Fu et al., 2026), with executable acceptance and repair in tool-use and embodied settings (Zhang et al., 2026; Jiang et al., 2026). Causal benchmarks use blueprint-constrained generation, consistency audits, and synthetic or semantic environments (Li et al., 2026b;a). Evaluation-set selection supports adversarial collection, score estimation, and analysis of task-choice effects (Kiela et al., 2021; Maia Polo et al., 2024; Dehghani et al., 2021).

3 BENCHSCOPE: EVIDENCE-GROUNDED BENCHMARK SCOPE ANALYSIS

BenchScope characterizes a benchmark's evaluation scope through structured boundary profiles of task conditions, generalization settings, protocol judgments, and evidence states (Figure 1). Dataset, target, split, and protocol records anchor these profiles to the tasks actually evaluated. An evidence-recovery agent traces and interprets heterogeneous sources, scripts verify factual quantities, and Boundary Skill applies explicit evidence requirements and decision rules to relate recovered conditions to the benchmark's stated objective.

3.1 RECOVERING BENCHMARK CONDITIONS FROM HETEROGENEOUS EVIDENCE

Benchmarks reuse source data with different targets, feature sets, splits, and preprocessing. Recovering the evaluation condition requires resolving the data identity and its task configuration. We distinguish a benchmark entry, a base-dataset identity, and a *task membership*: the inclusion of a dataset–target–split–protocol configuration in a benchmark. This representation links related evaluations while retaining their task-specific conditions.

The agent matches artifacts through source identifiers and uses name similarity to propose matches for review. It checks whether the paper, repository, and manifest describe the same evaluated data view. Metadata transfer across schemas requires a supported preprocessing mapping; target- and split-dependent values stay attached to task memberships.

Missing mappings, conflicting versions, and consequential evidence gaps trigger targeted reinspection, with human adjudication where needed. A supported resolution updates the record; otherwise the field retains its evidence status. Values are marked as observed, deterministically derived, inferred, or unknown and linked to source locations.

For example, the penguins audit finds different source and execution schemas. Without a recoverable preprocessing mapping, the agent retains the two views separately and leaves execution-level missingness and cardinality unknown.

3.2 BOUNDARY SKILL: FROM RECOVERED FACTS TO AUDITABLE JUDGMENTS

Boundary Skill determines what evidence is needed to interpret a recovered condition. Its 44 indicators span eight dimensions: identity and source diversity, scale and geometry, feature structure and data quality, prediction-task diversity, generalization and replication, comparison fairness, evaluation coverage, and result reliability and reproducibility; each rule specifies applicability, required evidence, and a computation or decision procedure. The agent applies these rules relative to the benchmark's stated objective. For example, IID coverage supports an in-distribution comparison, while a temporal-generalization claim requires temporal evaluation. This claim-to-condition check distinguishes specialized scope, missing coverage, and protocol conditions that conflict with the intended comparison.

The profile records coverage, regime occupancy, concentration, and protocol disclosures. Each coverage measure is paired with evidence completeness, separating observed coverage from the fraction of tasks on which it can be assessed.

For task-level predicate j, let A_j be applicable memberships, $K_j \subseteq A_j$ those with observed or derived values, and $P_j \subseteq K_j$ those satisfying it. We report both

$$c_j = \frac{|P_j|}{|K_j|}, \qquad e_j = \frac{|K_j|}{|A_j|}, \tag{1}$$

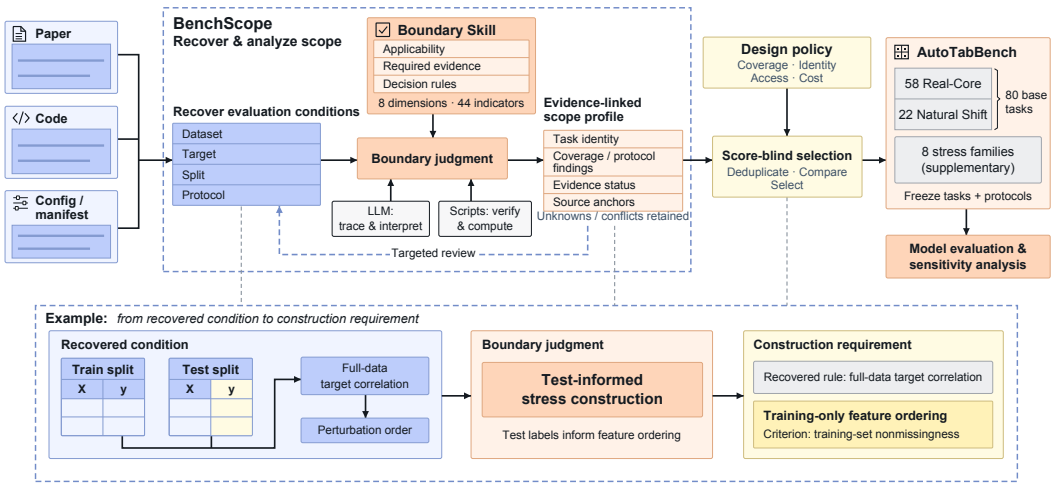

Figure 1: Boundary recovery and its application to benchmark design. Boundary Skill links evidence to coverage and protocol judgments; the example links a finding to training-only feature ordering.

where c_j measures coverage conditional on known values and e_j measures evidence completeness. A ratio with an empty denominator is undefined. Unknown and inferred values do not enter K_j; logically inapplicable memberships are excluded from A_j with a recorded rationale. The report retains counts and reasons for these exclusions alongside the ratios.

Regime-occupancy and concentration rules specify their own denominators. Versioned thresholds and bins allow sensitivity analysis on the same records; rules appear in Appendix B.5.

3.3 Structured Boundary Profiles

The Boundary Profile records counts, ranges, categorical values, and explicit evidence states, linked to task identities and source anchors. Coverage gaps identify additional test conditions, shared identities support deduplication, and protocol findings inform evaluation design (Section 5).

4 Cross-Benchmark Scope Analysis

We audit 37 benchmark entries from 30 sources, with benchmark-level cards for all entries and task-level inventories for 16, totaling 1,143 memberships. The inventories reveal complementary shift coverage: grouped and temporal evaluations in BeyondArena, domain shifts in TableShift, temporal prediction in TabReD (Rubachev et al., 2025), and feature shifts in TabFSBench. Three cases show how recovered conditions change their interpretation.

4.1 Findings that Change the Interpretation of an Evaluation

Test labels can enter the construction of a stress test. TabFSBench evaluates robustness to feature shifts (Cheng et al., 2025). Its implementation creates a train–test split but passes a copy of the original dataframe to the feature-ranking function, which computes full-data target correlations. Test labels therefore inform which features are perturbed. Tracing this input identifies the evaluation as robustness to target-informed perturbations, an information-access condition that a split description alone does not reveal. Figure 1 links this finding to our training-only stress ordering.

Score provenance determines what an aggregate compares. BeyondArena reports replacing missing TabPFN-2.6 and TabDPT results with default Random Forest performance (Purucker et al., 2026, Section 5, v1). Its aggregate combines direct model runs with substitute-model results. The audit records this provenance alongside execution status, distinguishing performance of the named model from performance under the fallback policy. This finding motivates preserving failed runs as separate outcomes in the new evaluation.

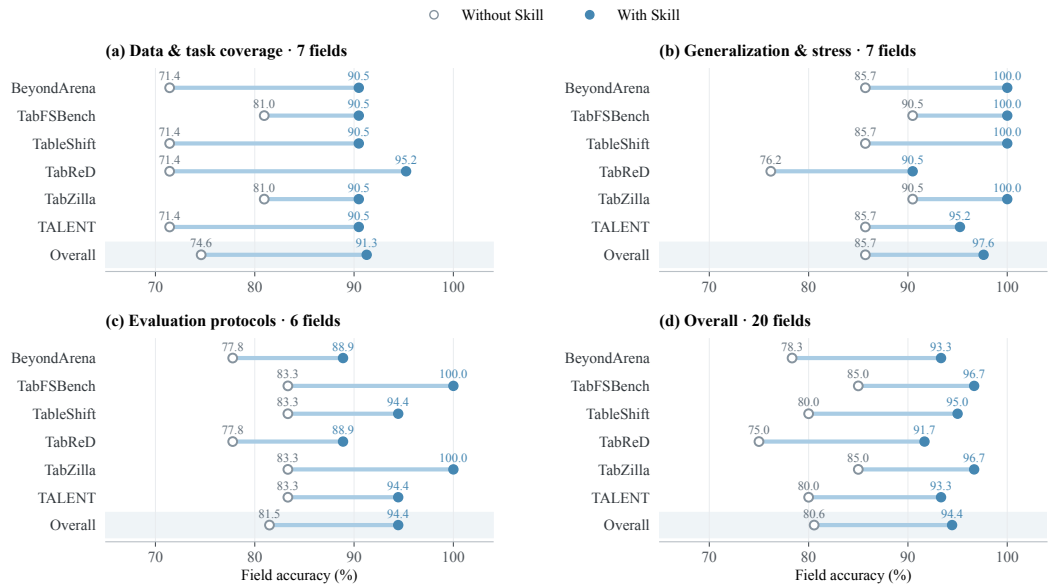

Figure 2: Boundary-profile accuracy by field group and benchmark. Points aggregate three runs.

Metadata must describe the artifact actually evaluated. The penguins decision in Section 3.1 changes the denominator of coverage reporting: a known source-level statistic does not qualify as a known executed-task value until the data views are reconciled. Missingness and categorical cardinality therefore remain evidence gaps.

4.2 Recovering boundary profiles and supporting evidence

We compare GPT-5.6 agents with and without Boundary Skill in isolated contexts, with identical papers, code, configurations, and access to file reading, web search, and code execution. Human-verified reference profiles comprise 20 fields covering data and tasks, generalization and stress conditions, and evaluation protocols. Six benchmarks with three runs per condition yield 36 profile runs. A separate 36-run claim-level evaluation measures recognition of 42 author-verified references, factual correctness, and evidence localization. Agent outputs are evaluated by the author, blinded to the Skill condition. Definitions and detailed results appear in Appendix B.

Boundary Skill raises profile accuracy from 80.56% to 94.44% (Table 1), with gains on all six benchmarks (Figure 2). Data and task coverage gains the most (+16.67 percentage points), while generalization and stress conditions are recovered most accurately (97.62%). Claim-level results show complementary gains: numerical-description errors fall from five to one, while reference recognition rises from 93/108 (86.1%) to 106/108 (98.1%). On the four external sources, citation failures fall from 45.3% to 19.4%, alongside recognition rising from 93.1% to 98.6%.

5 Scope-Guided Benchmark Construction

AutoTabBench is assembled by translating BenchScope profiles into three design decisions: which tasks are eligible, which provide complementary coverage, and how their protocols should be implemented.

5.1 Translating audit findings into constraints

Task identity and protocol recoverability provide the first selection criteria. Eligible candidates require identifiable data and targets, recoverable splits, executable metrics, and public access compatible with execution; confirmed target leakage and exact duplicate identities counted as independent evidence fail these checks. The policy uses 30% source-share and 20% domain-share targets to

Table 1: Boundary recovery and claim-level audit quality. Counts are pooled within each condition. Profile accuracy is higher-is-better; error and omission rates are lower-is-better.

Outcome	Without Skill	With Skill
Boundary-profile accuracy		
Data and task coverage	74.60% (94/126)	**91.27%** (115/126)
Generalization and stress	85.71% (108/126)	**97.62%** (123/126)
Evaluation protocols	81.48% (88/108)	**94.44%** (102/108)
All 20 fields	80.56% (290/360)	**94.44%** (340/360)
Claim-level audit quality		
Audit claims	173	201
Factual claim errors	2.89% (5/173)	**0.50%** (1/201)
Known-reference omissions	13.89% (15/108)	**1.85%** (2/108)
Designated-unknown omissions	50.00% (9/18)	**22.22%** (4/18)
Citation/line-check failures	37.50% (99/264)	**15.28%** (46/301)

limit concentration, and natural-shift quotas to preserve recovered shift coverage. These design constraints are fixed for candidate comparison, with coverage and concentration taking priority over the metadata-based computation proxy.

The protocol findings in Section 4.1 determine stress construction and outcome reporting. Feature-availability stress uses training-set nonmissingness in place of full-data target correlation, defining an availability-based ordering from training-observed information. The ledger preserves direct-run provenance and records out-of-memory, timeout, unsupported, and error outcomes separately.

Applying the eligibility policy to recovered evidence excluded nine configurations with unresolved task identity or target semantics, or a mismatch with the source objective. Recovering data-access and official split evidence also enabled 11 natural-shift tasks to enter the benchmark, extending grouped and temporal coverage.

5.2 Selecting complementary tasks under a fixed budget

The executable pool combines 80 previously validated configurations with eight TabArena candidates whose license, split, and domain evidence were recovered. It contains 88 distinct base identities. The broader inventory comprises 1,143 source-task memberships (Appendix C).

We compare 80-task collections using coverage, identity, evidence status, access, and a metadata-based cost proxy, excluding model scores, rankings, and execution failures from selection. Nine schemes emphasizing different coverage, concentration, and cost objectives produce seven distinct task sets under the same task-count budget.

The source target makes the selection problem concrete. The pool contains 32 TabArena candidates (Erickson et al., 2026) and 56 from other sources. An 80-task collection meeting the 30% source target can include at most 24 TabArena tasks, so it retains all 56 other candidates, including the 22 natural-shift tasks. Selection determines which 24 routine-prediction tasks from TabArena best complement this coverage under the priorities.

The policy selects coverage, concentration, and evidence criteria from the 44-indicator profiles and computes cost from task metadata. Pareto comparison identifies non-dominated collections, and the frozen priority order selects among them. Table 16 exposes the central trade-off: low-cost selection leaves a six-task shortfall against shift quotas, while domain balancing exceeds the source-share target. Both the selected and source-balanced collections meet the share targets and occupy the same numbers of sample-size, feature-count, and sample-to-feature-ratio bins. Prioritizing concentration selects the former, reducing domain HHI from 0.113 to 0.106 while increasing the metadata-based cost proxy from 974.5 to 989.1.

5.3 Final collection and executable protocol

AutoTabBench contains 58 Real-Core tasks for routine prediction and 22 Natural Shift tasks for generalization across grouped, temporal, and domain partitions: 13, five, and four tasks, respec-

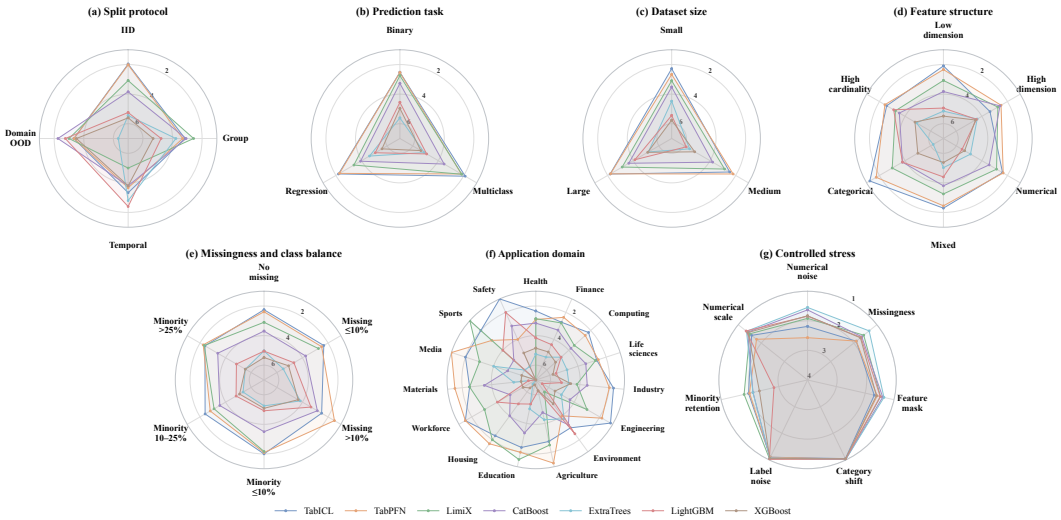

Figure 3: Own-success condition profiles. Outward is better: (a–f) mean rank, (g) median loss ratio. Gaps denote no successes.

tively (Table 17). Eight controlled-stress families supplement applicable Real-Core tasks to measure sensitivity to changes. Dataset identities link to task configurations for source-aware analysis; natural-shift and stress results retain separate denominators.

Before evaluation, the manifest fixes targets, features, splits, metrics, weights, and provenance. A reverse audit checks eligibility, duplicate identities, and compliance with source and domain targets, while retaining unresolved upstream licensing evidence in the task records. Section 6 evaluates this collection and tests sensitivity to the candidate mixtures.

6 Model Comparisons under Audited Evaluation Conditions

We evaluate seven models on AutoTabBench across routine, natural-shift, and stress conditions, assessing predictive performance, execution coverage, and perturbation robustness.

6.1 Evaluation protocol

We compare TabPFN, TabICL, and LimiX (Grinsztajn et al., 2026; QU et al., 2026; Zhang et al., 2025) with CatBoost, XGBoost, LightGBM, and ExtraTrees (Prokhorenkova et al., 2018; Chen & Guestrin, 2016; Ke et al., 2017; Geurts et al., 2006) under the frozen default/zero-shot protocol. All models use three seeds without task-specific hyperparameter search. Classification uses log-loss and regression uses RMSE. All models receive shared numerical arrays with training-fitted ordinal category codes; CatBoost, LightGBM, and XGBoost treat these codes as numerical inputs. Stress is applied before preprocessing.

All 1,680 base task–model–seed cells have recorded terminal outcomes; 1,584 succeed. We average successful-seed losses within each task, rank the available models with average ranks for ties, and average task ranks. *Own-success* results use each model's successful tasks, with execution counts reported alongside them. *Failure-last* ranking places models lacking three successful seeds jointly after complete models, using the average of their occupied rank positions. For paired comparisons, the *common-success* subset requires all seven models to complete all three seeds.

6.2 Aggregate leaders change across evaluation conditions

TabICL and TabPFN lead the 80-task aggregate under both reporting rules, with failure-last ranks of 2.46 and 2.59 (Table 2). On Real-Core, their own-success ranks are 1.965 and 2.018. Their perfor-

Table 2: Performance and execution by track. S: successful cells; Own: own-success mean rank; FL: failure-last rank. Lower ranks are better.

	Overall (80 tasks)			Real-Core (58 tasks)			Natural Shift (22 tasks)		
Model	S/240	Own	FL	S/174	Own	FL	S/66	Own	FL
TabICL	228	**2.28**	**2.46**	171	**1.96**	**2.03**	57	3.21	3.57
TabPFN	225	2.36	2.59	168	2.02	2.16	57	3.37	3.70
LimiX	174	3.05	4.08	147	3.08	3.66	27	**2.89**	5.16
CatBoost	240	3.64	3.64	174	3.84	3.84	66	3.09	**3.09**
ExtraTrees	237	5.04	5.04	174	5.45	5.45	63	3.90	3.98
LightGBM	240	4.86	4.86	174	5.24	5.24	66	3.86	3.86
XGBoost	240	5.34	5.34	174	5.60	5.60	66	4.64	4.64

Table 3: Numerical noise at one training IQR on 46 common-success tasks. Lower is better; ratios are medians of task-wise paired loss ratios.

Model	Unperturbed rank	Stressed rank	Loss ratio
TabICL	**2.043**	3.674	1.707
TabPFN	2.087	3.739	2.100
LimiX	3.087	**2.804**	1.917
CatBoost	3.957	3.761	1.631
ExtraTrees	5.478	2.870	**1.548**
LightGBM	5.457	4.978	1.753
XGBoost	5.891	6.174	1.829

mance is closely matched on the 49 jointly completed Real-Core tasks: the paired rank difference is -0.102, with a descriptive task-bootstrap 95% interval of $[-0.571, 0.388]$.

Natural Shift changes both the leading model and the importance of execution coverage. LimiX has the lowest own-success rank (2.889), but completes 27 of 66 cells; CatBoost completes all 66 and leads under failure-last ranking (3.091), while LimiX moves to 5.159. Restricting the comparison to nine jointly completed tasks instead places TabICL and LimiX together at 2.889. LimiX's sample/class guards, memory failures, and CUDA errors are retained in the ledger. Successful-task and full-track performance support different model choices.

The condition profiles locate distinctions hidden by these track averages (Figure 3). LightGBM has the lowest own-success rank on the five temporal tasks (2.40), while CatBoost leads the four domain-OOD tasks (2.25); both complete all tasks in the respective groups. High-dimensional tasks also favor a different comparison: TabPFN ranks 2.50 on ten successful tasks and CatBoost 2.58 on all 12, versus TabICL's 3.36 on 11.

6.3 Stress separates predictive rank from relative degradation

Numerical noise reverses the prediction advantage on a fixed set of 46 tasks for which seven models complete three seeds before and after perturbation. At one training IQR, TabICL and TabPFN move from unperturbed ranks 2.043 and 2.087 to 3.674 and 3.739; LimiX and ExtraTrees lead at 2.804 and 2.870 (Table 3). ExtraTrees has a smaller median loss ratio than LimiX (1.548 versus 1.917), combining competitive stressed performance with less degradation.

The paired rank difference between TabICL and LimiX shifts by 1.913 positions toward LimiX, with a descriptive task-bootstrap 95% interval of $[1.087, 2.804]$; Table 4 reports the highlighted pairs.

Table 4: Paired rank-difference changes under numerical noise. Negative unperturbed differences favor A; positive changes favor B.

A	B	Unperturbed	Noise	Change	95% interval
TabICL	LimiX	-1.043	0.870	1.913	[1.087, 2.804]
TabICL	ExtraTrees	-3.435	0.804	4.239	[3.239, 5.217]
TabPFN	LimiX	-1.000	0.935	1.935	[1.152, 2.761]
TabPFN	ExtraTrees	-3.391	0.870	4.261	[3.391, 5.174]

Stress-Grid records 25,473 successes across 26,166 cells and 1,246 task–condition combinations. Figure 4 reports two complementary outcomes: mean rank of stressed loss and median stressed-to-unperturbed loss ratio. Both use tasks with three successful unperturbed–stress seed pairs per model and condition; ratios are computed from three-seed mean losses. Ranks compare available paired-success models; task subsets and competitor counts may differ. A ratio of 1 denotes no change.



Figure 4: Own-success paired stress results: mean rank (pink circles) and median loss ratio (yellow diamonds), both lower-is-better.

The broader stress profiles separate predictive performance from relative degradation. LimiX has the lowest own-success mean rank under minority retention, missingness, numerical noise, and scale expansion. TabPFN leads under unseen-category and label perturbations and nearly ties LimiX under feature masking (3.036 versus 3.041). ExtraTrees has the smallest median loss ratio under feature masking, missingness, numerical noise, and unseen-category stress. The rank identifies strong predictors under each stress; the ratio identifies models preserving more unperturbed performance.

6.4 Composition stability and reporting sensitivity

TabICL remains first under failure-last ranking in all seven distinct collections produced by the nine selection schemes. Each collection selects 80 of 88 candidates, so any pair shares at least 72 tasks; this assesses local sensitivity to composition within the audited pool. Aggregation choices have a larger effect. Across 132 settings varying scope, weighting, group omission, and task or cost budgets, 17 change the first-ranked model and 49 change at least one pairwise order (Table 5). Under failure-last ranking, sample weighting places LightGBM first, whereas equal weighting of tasks, sources, or domains favors TabICL. Thus the locally stable task-selection result coexists with sensitivity to which prediction settings receive the most weight.

Table 5: Composition and reporting sensitivity. Details are shown in Table 24.

Analysis	Scale	Main result
Task collections	7 distinct 80-task sets	TabICL first in all
Collection overlap	Pairwise collections	At least 72 shared tasks
Reporting settings	132 aggregation settings	17 change the first-ranked model
Pairwise orders	132 aggregation settings	49 change at least one pairwise order

7 Conclusion

BenchScope characterizes benchmark scope through evidence-linked profiles of task coverage, generalization conditions, evaluation protocols, and evidence states. Boundary Skill improves structured boundary recovery from 80.56% to 94.44% across six benchmarks, alongside gains in reference recognition and evidence localization. Applying BenchScope to 37 benchmark entries, we construct AutoTabBench, whose routine-prediction, natural-shift, and controlled-stress evaluations show that model advantages depend on evaluation scope.

AI USE STATEMENT

During manuscript preparation, AI tools were used for language polishing and improving readability. The authors reviewed and edited the text and take full responsibility for the final manuscript.

REPRODUCIBILITY STATEMENT

Appendices A–B describe the audit rules, reference profiles, scoring criteria, and run-level results. Appendix C details benchmark construction, and Appendices D–E summarize the audit corpus and coverage. Appendix F specifies model configurations, preprocessing, stress transformations, and additional evaluation results.

REFERENCES

Alan Arazi, Eilam Shapira, and Roi Reichart. Tabstar: A tabular foundation model for tabular data with text fields, 2025. URL https://arxiv.org/abs/2505.18125.

Alan Arazi, Eilam Shapira, Shoham Grunblat, Mor Ventura, Elad Hoffer, Gioia Blayer, David Holzmüller, Lennart Purucker, Gaël Varoquaux, Frank Hutter, and Roi Reichart. Multabench: Benchmarking multimodal tabular learning with text and image, 2026. URL https://arxiv.org/abs/2605.10616.

Bernd Bischl, Giuseppe Casalicchio, Matthias Feurer, Pieter Gijsbers, Frank Hutter, Michel Lang, Rafael G. Mantovani, Jan N. van Rijn, and Joaquin Vanschoren. Openml benchmarking suites, 2021. URL https://arxiv.org/abs/1708.03731.

Natasha Butt, Varun Chandrasekaran, Neel Joshi, Besmira Nushi, and Vidhisha Balachandran. Benchagents: Automated benchmark creation with agent interaction. In *ICLR 2025 Workshop on Navigating and Addressing Data Problems for Foundation Models*, 2025. URL https://openreview.net/forum?id=Xh6S3X3enu.

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*, KDD '16, pp. 785–794, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939785. URL https://doi.org/10.1145/2939672.2939785.

Zi-Jian Cheng, Zi-Yi Jia, Zhi Zhou, Yu-Feng Li, and Lan-Zhe Guo. Tabfsbench: tabular benchmark for feature shifts in open environments. ICML'25. JMLR.org, 2025.

DataCebo. SDGym: Synthetic data gym documentation. https://docs.sdv.dev/sdgym. Accessed: 2026-09-22.

José Lucas De Melo Costa, Fabrice Popineau, Arpad Rimmel, and Bich-Liên Doan. High performance, low reliability: Uncertainty benchmarking for tabular foundation models. In *ESANN 2026 proceesdings*, ESANN 2026, pp. 115–120. Ciaco - i6doc.com, 2026. doi: 10.14428/esann/2026.es2026-261. URL http://dx.doi.org/10.14428/esann/2026.ES2026-261.

Mostafa Dehghani, Yi Tay, Alexey A. Gritsenko, Zhe Zhao, Neil Houlsby, Fernando Diaz, Donald Metzler, and Oriol Vinyals. The benchmark lottery, 2021. URL https://arxiv.org/abs/2107.07002.

Nick Erickson, Lennart Purucker, Andrej Tschalzev, David Holzmüller, Prateek Mutalik Desai, David Salinas, and Frank Hutter. Tabarena: A living benchmark for machine learning on tabular data. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2026. URL https://openreview.net/forum?id=jZqCqpCLdU.

Sebastian Felix Fischer, Matthias Feurer, and Bernd Bischl. OpenML-CTR23 – a curated tabular regression benchmarking suite. In *AutoML Conference 2023 (Workshop)*, 2023. URL https://openreview.net/forum?id=HebAOoMm94.

Lingyue Fu, Bolun Zhang, Hao Guan, Yaoming Zhu, Lin Qiu, Weiwen Liu, Xuezhi Cao, Xunliang Cai, Weinan Zhang, and Yong Yu. Automatically benchmarking llm code agents through agent-driven annotation and evaluation. In *Proceedings of the 25th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '26, pp. 874–882, Richland, SC, 2026. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9798400723179. doi: 10.65109/HJFB4234. URL https://doi.org/10.65109/HJFB4234.

Josh Gardner, Zoran Popović, and Ludwig Schmidt. Benchmarking distribution shift in tabular data with tableshift. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Joshua P Gardner, Juan Carlos Perdomo, and Ludwig Schmidt. Large scale transfer learning for tabular data via language modeling. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=WH5blx5tZ1.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. Datasheets for datasets. *Commun. ACM*, 64(12):86–92, November 2021. ISSN 0001-0782. doi: 10.1145/3458723. URL https://doi.org/10.1145/3458723.

Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Mach. Learn.*, 63(1):3–42, April 2006. ISSN 0885-6125. doi: 10.1007/s10994-006-6226-1. URL https://doi.org/10.1007/s10994-006-6226-1.

Pieter Gijsbers, Marcos L. P. Bueno, Stefan Coors, Erin LeDell, Sébastien Poirier, Janek Thomas, Bernd Bischl, and Joaquin Vanschoren. Amlb: an automl benchmark. *J. Mach. Learn. Res.*, 25 (1), January 2024. ISSN 1532-4435.

Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2021. Curran Associates Inc. ISBN 9781713845393.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Mihir Manium, Shi Bin Hoo, Magnus Bühler, Anurag Garg, Dominik Safaric, Jake Robertson, Benjamin Jäger, Simone Alessi, Adrian Hayler, Vladyslav Moroshan, Lennart Purucker, Philipp Singer, Alan Arazi, Julien Siems, Jan Hendrik Metzen, Georg Grab, Nick Erickson, Siyuan Guo, Eliott Kalfon, Simon Bing, David Salinas, Clara Cornu, Lilly Charlotte Wehrhahn, Diana Kriuchkova, Kursat Kaya, Lydia Sidhoum, Marie Salmon, Jerry Chen, Madelon Hulsebos, Yann LeCun, Samuel Müller, Bernhard Schölkopf, Sauraj Gambhir, Noah Hollmann, and Frank Hutter. Tabpfn-3: Technical report, 2026. URL https://arxiv.org/abs/2605.13986.

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(8045):319–326, 2025. ISSN 1476-4687. doi: 10.1038/s41586-024-08328-6. URL https://doi.org/10.1038/s41586-024-08328-6.

Rafael Izbicki and Pedro L. C. Rodrigues. Benchmarking tabular foundation models for conditional density estimation in regression, 2026. URL https://arxiv.org/abs/2603.26611.

Baoyang Jiang, Fengchun Zhang, Leyuan Wang, Haotian Li, Yida Wang, Zhe Ji, Jinshan Lai, Xi Ren, Danyang Li, Zheng Yang, Jianwei Hu, and Qiang Ma. Embodied-BenchForge: A Closed-Loop Agentic Workflow for Embodied Benchmark Construction, 2026. URL https://arxiv.org/abs/2609.13082.

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 *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 3149–3157, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. Dynabench: Rethinking benchmarking in NLP. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4110–4124, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.324. URL https://aclanthology.org/2021.naacl-main.324/.

Ricardo Knauer, Marvin Grimm, and Erik Rodner. Pmlbmini: A tabular classification benchmark suite for data-scarce applications, 2024. URL https://arxiv.org/abs/2409.01635.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran Haque, Sara M Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5637–5664. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/koh21a.html.

Xiang Lisa Li, Farzaan Kaiyom, Evan Zheran Liu, Yifan Mai, Percy Liang, and Tatsunori Hashimoto. Autobencher: Towards declarative benchmark construction, 2025. URL https://arxiv.org/abs/2407.08351.

Zi-Rong Li, Si-Yang Liu, Tian-Zuo Wang, and Han-Jia Ye. Causalarena: Benchmarking causal discovery in the foundation model era, 2026a. URL https://arxiv.org/abs/2609.11897.

Zi-Rong Li, Si-Yang Liu, Tian-Zuo Wang, and Han-Jia Ye. Causaltab: Pretraining across causal environments for tabular causal discovery. In *2nd ICML Workshop on Foundation Models for Structured Data*, 2026b. URL https://openreview.net/forum?id=og3UVhP7M1.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D. Manning, Christopher Ré, Diana Acosta-Navas, Drew A. Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic evaluation of language models, 2023. URL https://arxiv.org/abs/2211.09110.

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 http://jmlr.org/papers/v26/25-0512.html.

Malena Loza, David Chushig-Muzo, Eva Milara, Luis Bote-Curiel, Luis Estrada-Petrocelli, and Felipe Grijalva. Empirical evaluation of out-of-distribution performance of tabular foundation models, 2026. URL https://arxiv.org/abs/2607.26000.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinyBenchmarks: evaluating LLMs with fewer examples. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 34303–34326. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/maia-polo24a.html.

Duncan McElfresh, Sujay Khandagale, Jonathan Valverde, Vishak Prasad C., Ganesh Ramakrishnan, Micah Goldblum, and Colin White. When do neural nets outperform boosted trees on tabular

data? In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pp. 6639–6649, Red Hook, NY, USA, 2018. Curran Associates Inc.

Lennart Purucker, Andrej Tschalzev, Nick Erickson, Gioia Blayer, David Holzmüller, Alan Arazi, Alexander Pfefferle, Mustafa Tajjar, Gaël Varoquaux, and Frank Hutter. Beyond iid: How general are tabular foundation models, really?, 2026. URL https://arxiv.org/abs/2606.30410.

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*, ICML'25. JMLR.org, 2025.

Jingang QU, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICLv2: A better, faster, scalable, and open tabular foundation model. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=SxsyLjIfWB.

Joseph D Romano, Trang T Le, William La Cava, John T Gregg, Daniel J Goldberg, Praneel Chakraborty, Natasha L Ray, Daniel Himmelstein, Weixuan Fu, and Jason H Moore. Pmlb v1.0: an open-source dataset collection for benchmarking machine learning methods. *Bioinformatics*, 38(3):878–880, 02 2022. ISSN 1367-4803. doi: 10.1093/bioinformatics/btab727. URL https://doi.org/10.1093/bioinformatics/btab727.

Ivan Rubachev, Nikolay Kartashev, Yury Gorishniy, and Artem Babenko. Tabred: Analyzing pitfalls and filling the gaps in tabular deep learning benchmarks. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=L14sqcrUC3.

David Salinas and Nick Erickson. Tabrepo: A large scale repository of tabular model evaluations and its automl applications. In Katharina Eggensperger, Roman Garnett, Joaquin Vanschoren, Marius Lindauer, and Jacob R. Gardner (eds.), *Proceedings of the Third International Conference on Automated Machine Learning*, volume 256 of *Proceedings of Machine Learning Research*, pp. 19/1–30. PMLR, 09–12 Sep 2024. URL https://proceedings.mlr.press/v256/salinas24a.html.

Xingjian Shi, Jonas Mueller, Nick Erickson, Mu Li, and Alexander J. Smola. Benchmarking multimodal automl for tabular data with text fields, 2021. URL https://arxiv.org/abs/2111.02705.

Dimitrios Sinodinos, Jack Yi Wei, and Narges Armanfard. Multitab: a scalable foundation for multitask learning on tabular data. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence and Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence and Sixteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'26/IAAI'26/EAAI'26. AAAI Press, 2026. ISBN 978-1-57735-906-7. doi: 10.1609/aaai.v40i30.39745. URL https://doi.org/10.1609/aaai.v40i30.39745.

Dylan Slack and Sameer Singh. Tablet: Learning from instructions for tabular data, 2023. URL https://arxiv.org/abs/2304.13188.

Zhiqiang Tang, Zihan Zhong, Tong He, and Gerald Friedland. Bag of tricks for multimodal automl with image, text, and tabular data, 2024. URL https://arxiv.org/abs/2412.16243.

Quan M. Tran, Suong N. Hoang, Lam M. Nguyen, Dzung Phan, and Hoang Thanh Lam. Tabularfm: An open framework for tabular foundational models, 2024. URL https://arxiv.org/abs/2406.09837.

Xinming Tu, Tianze Wang, Yingzhou Lu, Kexin Huang, Yuanhao Qu, and Sara Mostafavi. Who guards the benchmarks? automated auditing of LLM agent benchmarks. In *Third Conference on Language Modeling*, 2026. URL https://openreview.net/forum?id=3x89U8nvoE.

Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luis Torgo. Openml: networked science in machine learning. *SIGKDD Explor. Newsl.*, 15(2):49–60, June 2014. ISSN 1931-0145. doi: 10.1145/2641190.2641198. URL https://doi.org/10.1145/2641190.2641198.

Junlin Wang, Federico Bianchi, Shang Zhu, Fan Nie, Yongchan Kwon, Bhuwan Dhingra, and James Zou. Automated benchmark auditing for ai agents and large language models, 2026. URL https://arxiv.org/abs/2605.26079.

Jiahuan Yan, Bo Zheng, Hongxia Xu, Yiheng Zhu, Danny Z. Chen, Jimeng Sun, Jian Wu, and Jintai Chen. Making pre-trained language models great on tabular prediction, 2024. URL https://arxiv.org/abs/2403.01841.

Ke Zhang, Yankang Liu, Roya Zandi, and Maziar Raissi. Toolgate: An executable acceptance pipeline for tool-dependent scientific benchmark construction, 2026. URL https://arxiv.org/abs/2609.02067.

Xingxuan Zhang, Gang Ren, Han Yu, Hao Yuan, Hui Wang, Jiansheng Li, Jiayun Wu, Lang Mo, Li Mao, Mingchao Hao, Ningbo Dai, Renzhe Xu, Shuyang Li, Tianyang Zhang, Yue He, Yuanrui Wang, Yunjia Zhang, Zijing Xu, Dongzhe Li, Fang Gao, Hao Zou, Jiandong Liu, Jiashuo Liu, Jiawei Xu, Kaijie Cheng, Kehan Li, Linjun Zhou, Qing Li, Shaohua Fan, Xiaoyu Lin, Xinyan Han, Xuanyue Li, Yan Lu, Yuan Xue, Yuanyuan Jiang, Zimu Wang, Zhenlei Wang, and Peng Cui. Limix: Unleashing structured-data modeling capability for generalist intelligence, 2025. URL https://arxiv.org/abs/2509.03505.

A Audit Records and Selection Policy

A.1 Evidence provenance and rule revisions

Recovered fields retain evidence locations, source versions or content hashes when available, and value status. Directly observed facts, deterministically derived values, and semantic inferences are distinguished; source reliability is recorded separately from the operation used to obtain a value. Conflicting statements retain competing evidence and resolution status, including differences between paper descriptions and executable configurations. Unknown fields record why evidence is unavailable.

The Boundary Skill formation ledger traces the initial indicators to recurring questions in the designated source materials. Each indicator contract defines its analysis unit, applicability, required evidence, computation or decision procedure, and treatment of missing or conflicting information. Evidence corrections update records; changes in meaning, applicability, or calculation produce a rule revision and a separately identified profile version. This preserves the interpretation under which each profile was computed.

Evidence status distinguishes conditions unspecified by inspected sources, such as upstream licensing in BeyondArena, from task-level conditions awaiting expansion of a benchmark-level audit. The former calls for source evidence; the latter calls for task-level examination.

A.2 Frozen score-blind selection order

Hard eligibility checks require an identifiable dataset, an explicit classification or regression target, a recoverable split, an executable metric, and public access compatible with the intended execution. They exclude confirmed target leakage and exact duplicate identities counted as independent evidence. Candidate collections are compared by Pareto dominance over the declared design objectives. The selector applies the following lexicographic order to the non-dominated alternatives, preferring the direction shown at the first differing criterion.

Order	Criterion	Preference
1	Hard-constraint failure	Minimize
2	Natural-shift requirement shortfall	Minimize
3	Duplicate identities	Minimize
4	Excess above the 30% source-share target	Minimize
5	Excess above the 20% domain-share target	Minimize
6	Tasks with unknown domain	Minimize
7	License-record shortfall	Minimize
8	Occupied sample-size bins	Maximize
9	Occupied feature-count bins	Maximize
10	Occupied sample-to-feature-ratio bins	Maximize
11	Source HHI	Minimize
12	Domain HHI	Minimize
13	Metadata-based computation proxy	Minimize
14	Scheme name (deterministic tie-break)	Lexical order

Selection uses the metadata-based computation criterion and candidate-registry license entries. Model scores, rankings, execution failures, and measured runtime are excluded; upstream licensing uncertainties remain in the task-level records.

A.3 Worked audit decision and case summary

A worked decision: resolving a data-view mismatch. The penguins CSV contains 344 rows and five columns, while the repository's test-feature array has shape 69×6. Establishing their relationship requires the preprocessing transformation and full training/validation payload, which the saved evidence does not recover. The audit keeps both views and marks the transfer of missingness and maximum categorical cardinality as `conflicting_evidence`. Source-view statistics remain available; the execution-level fields remain unknown.

Table 6: Recovered conditions and their consequences for evaluation design.

Audit object	Recovered evidence	Interpretation	Evaluation decision
Perturbation construction	Feature order uses full-data target correlations	Test labels inform stress design	Use training-only statistics for feature-availability stress
Score provenance	Missing runs receive Random Forest scores	Aggregate mixes direct and substitute-model results	Retain failures and direct-run provenance
Data view	Linked CSV and execution array differ; mapping unresolved	Task-level statistics remain an evidence gap	Keep views separate; withhold unsupported transfer
Temporal scope	Temporal splits match a time-dependent prediction objective	Legitimate specialization	Preserve the stated scope

B Boundary Skill and Controlled Audit Evaluation

B.1 Reference materials and experimental design

The six benchmarks are TabZilla, TALENT, TabReD, TableShift, TabFSBench, and BeyondArena. TabZilla and TALENT are indicator-formation sources; the other four are external-audit sources (Table 18). Both conditions use fresh GPT-5.6 contexts, identical papers, code, and configurations, and access to file reading, web search, and code execution. The Skill condition additionally receives the frozen 44-indicator definitions and evidence-handling rules.

B.2 Boundary-profile recovery

We represent each benchmark by 20 fields spanning data and task coverage (seven fields), generalization and stress conditions (seven), and evaluation protocols (six). Accuracy is the fraction of these 20 fields exactly matching the human-verified reference, including unknown and not-applicable answers; ranges require both endpoints to match, and set-valued answers require matching sets. Three runs on each of six benchmarks yield 360 field comparisons per condition.

B.3 Claim-level scoring and aggregation

The author-verified reference checklist, fixed before the claim-level experiment, contains six supported boundary judgments and one insufficient-evidence judgment per benchmark (42 in total). Three repetitions per condition yield 108 known-reference and 18 unknown-reference opportunities per condition. The author evaluates all 36 agent outputs, blinded to the Skill condition.

Recognition requires an output to substantively express the reference condition; recognizing an unknown requires explicitly identifying insufficient evidence. Omission rates divide missed references by 108 known-reference or 18 unknown-reference opportunities per condition. Recognition is the complementary rate. Equal reference counts make pooled and mean per-run recognition identical.

Factual errors are claims judged incorrect against the source materials. False absence marks the specific error of interpreting missing documentation as absence of a capability. Reference recall counts recognized checklist items; additional correct claims contribute to the output total.

Factual error rates use output claims as their denominator; citation failures use individual citations, including multiple citations attached to one claim. Citation checks assess quoted text and source locations. Rates pool counts across runs (Table 1).

Table 7: The 20-field benchmark boundary profile. All fields also admit unknown and not-applicable states. Counts refer to distinct dataset–target–split tasks, excluding repeated runs.

Field	Condition	Answer format
Data and task coverage		
1	Classification tasks	Count
2	Regression tasks	Count
3	Training samples per task	Minimum and maximum
4	Input features per task	Minimum and maximum
5	Classes per classification task	Minimum and maximum
6	Tasks containing categorical features	Count
7	Tasks containing missing inputs	Count
Generalization and stress conditions		
8	Tasks using random splits	Count
9	Tasks using temporal splits	Count
10	Tasks with test groups absent from training	Count
11	Tasks with test domains absent from training	Count
12	Feature-perturbation tests	Yes / no
13	Training-label perturbation tests	Yes / no
14	Training-sample reduction tests	Yes / no
Evaluation protocols		
15	Identical test samples across models	Yes / no / partial
16	Preprocessing fitting scope	Training only / includes validation / includes test / mixed
17	Test labels used for model or hyperparameter selection	Yes / no / partial
18	Test labels used in perturbation construction	Yes / no / partial
19	Substitute-model scores in aggregates	Yes / no
20	Failed-run handling	Omission / fixed penalty / failure-last / substitute model / whole-task removal

Table 8: Profile recovery by benchmark: three runs and 60 field comparisons per condition.

Benchmark	Without Skill	With Skill	Gain (pp)
BeyondArena	78.33% (47/60)	93.33% (56/60)	+15.00
TabFSBench	85.00% (51/60)	96.67% (58/60)	+11.67
TableShift	80.00% (48/60)	95.00% (57/60)	+15.00
TabReD	75.00% (45/60)	91.67% (55/60)	+16.67
TabZilla	85.00% (51/60)	96.67% (58/60)	+11.67
TALENT	80.00% (48/60)	93.33% (56/60)	+13.33
Overall	80.56% (290/360)	94.44% (340/360)	+13.89

Table 9: Boundary-profile run scores. Group counts follow data/tasks, generalization/stress, and protocols, with maxima 7, 7, and 6. Run numbers identify repetitions.

		Without Skill		With Skill	
Benchmark	Run	Group counts	Total	Group counts	Total
BeyondArena	1	5, 6, 4	15/20	6, 7, 5	18/20
BeyondArena	2	5, 6, 5	16/20	6, 7, 6	19/20
BeyondArena	3	5, 6, 5	16/20	7, 7, 5	19/20
TabFSBench	1	6, 6, 5	17/20	6, 7, 6	19/20
TabFSBench	2	5, 6, 5	16/20	6, 7, 6	19/20
TabFSBench	3	6, 7, 5	18/20	7, 7, 6	20/20
TableShift	1	5, 6, 5	16/20	6, 7, 6	19/20
TableShift	2	5, 6, 4	15/20	7, 7, 6	20/20
TableShift	3	5, 6, 6	17/20	6, 7, 5	18/20
TabReD	1	5, 5, 4	14/20	7, 6, 5	18/20
TabReD	2	5, 6, 5	16/20	7, 7, 5	19/20
TabReD	3	5, 5, 5	15/20	6, 6, 6	18/20
TabZilla	1	5, 6, 5	16/20	6, 7, 6	19/20
TabZilla	2	6, 7, 5	18/20	7, 7, 6	20/20
TabZilla	3	6, 6, 5	17/20	6, 7, 6	19/20
TALENT	1	5, 6, 5	16/20	6, 7, 6	19/20
TALENT	2	5, 6, 4	15/20	6, 6, 6	18/20
TALENT	3	5, 6, 6	17/20	7, 7, 5	19/20

B.4 Benchmark-specific and run-level Skill results

Tables 10–12 report the blind-scored comparison at benchmark, error-type, and run level. Counts are pooled within each condition. Known-reference and designated-unknown denominators are 18 and three per benchmark, respectively. Citation failures have their own citation-level denominator.

Table 10: Benchmark-level audit counts, without Skill → with Skill. Errors use claims; known and unknown omissions use reference opportunities; citation failures use citations.

Benchmark	Errors/claims	Known missed/18	Unknown missed/3	Citations failed
BeyondArena	$0/34 \rightarrow 0/33$	$0 \rightarrow 0$	$3 \rightarrow 3$	$42/62 \rightarrow 18/47$
TabFSBench	$0/32 \rightarrow 0/30$	$0 \rightarrow 0$	$1 \rightarrow 0$	$14/41 \rightarrow 4/49$
TableShift	$0/24 \rightarrow 0/32$	$2 \rightarrow 0$	$2 \rightarrow 1$	$17/37 \rightarrow 7/47$
TabReD	$4/24 \rightarrow 1/22$	$3 \rightarrow 1$	$1 \rightarrow 0$	$8/39 \rightarrow 6/37$
TabZilla	$1/29 \rightarrow 0/40$	$3 \rightarrow 0$	$1 \rightarrow 0$	$11/41 \rightarrow 5/55$
TALENT	$0/30 \rightarrow 0/44$	$7 \rightarrow 1$	$1 \rightarrow 0$	$7/44 \rightarrow 6/66$

TabZilla and TALENT are indicator-formation sources; the other four benchmarks are external-audit sources. Known-reference recognition rises from 72.2% to 97.2% on formation sources and from 93.1% to 98.6% on external sources. Nine of the 13 additional recognized reference opportunities come from formation sources. External-source citation failures fall from 45.3% to 19.4%, showing improved evidence localization alongside a smaller recall gain from a higher initial recognition rate.

The largest reduction in missed known references occurs on TALENT (seven to one); TabFSBench and BeyondArena already recover all known references in both conditions. Designated-unknown omissions persist in all three BeyondArena runs, while falling from two to one for TableShift and to zero for the remaining benchmarks. Citation failures decrease on every benchmark.

Factual errors concern numerical descriptions of the audited materials. A citation failure is assessed separately and does not automatically imply a factual error. False absence denotes treating missing documentation as evidence that a capability is absent.

Table 11: Factual-error types and false-absence counts across 18 runs per condition.

Type	Without Skill	With Skill
Dataset-total miscount	1	0
Classification/regression composition miscount	2	1
Sample-count disclosure miscount	2	0
Total factual errors	5	1
False absence	0	0

Table 12: All 36 audit runs. E/C: factual errors/claims; K: known omissions/6; U: designated-unknown omissions/1; F/T: failed/total citations.

Run	Skill	Benchmark	Claims	E/C	K	U	F/T
1	No	BeyondArena	11	0/11	0/6	1/1	3/18
2	No	BeyondArena	12	0/12	0/6	1/1	22/24
3	No	BeyondArena	11	0/11	0/6	1/1	17/20
4	No	TabFSBench	10	0/10	0/6	0/1	5/14
5	No	TabFSBench	9	0/9	0/6	1/1	5/13
6	No	TabFSBench	13	0/13	0/6	0/1	4/14
7	No	TableShift	7	0/7	1/6	1/1	6/11
8	No	TableShift	10	0/10	0/6	1/1	6/14
9	No	TableShift	7	0/7	1/6	0/1	5/12
10	No	TabReD	9	0/9	1/6	0/1	2/15
11	No	TabReD	7	2/7	1/6	1/1	4/12
12	No	TabReD	8	2/8	1/6	0/1	2/12
13	No	TabZilla	10	0/10	1/6	0/1	4/14
14	No	TabZilla	8	1/8	1/6	1/1	4/13
15	No	TabZilla	11	0/11	1/6	0/1	3/14
16	No	TALENT	11	0/11	2/6	0/1	1/13
17	No	TALENT	11	0/11	2/6	0/1	3/19
18	No	TALENT	8	0/8	3/6	1/1	3/12
19	Yes	BeyondArena	9	0/9	0/6	1/1	5/17
20	Yes	BeyondArena	12	0/12	0/6	1/1	4/16
21	Yes	BeyondArena	12	0/12	0/6	1/1	9/14
22	Yes	TabFSBench	11	0/11	0/6	0/1	2/20
23	Yes	TabFSBench	9	0/9	0/6	0/1	1/13
24	Yes	TabFSBench	10	0/10	0/6	0/1	1/16
25	Yes	TableShift	10	0/10	0/6	1/1	2/16
26	Yes	TableShift	11	0/11	0/6	0/1	3/16
27	Yes	TableShift	11	0/11	0/6	0/1	2/15
28	Yes	TabReD	7	0/7	0/6	0/1	2/13
29	Yes	TabReD	9	0/9	0/6	0/1	2/16
30	Yes	TabReD	6	1/6	1/6	0/1	2/8
31	Yes	TabZilla	8	0/8	0/6	0/1	0/12
32	Yes	TabZilla	18	0/18	0/6	0/1	3/23
33	Yes	TabZilla	14	0/14	0/6	0/1	2/20
34	Yes	TALENT	15	0/15	1/6	0/1	2/22
35	Yes	TALENT	13	0/13	0/6	0/1	1/20
36	Yes	TALENT	16	0/16	0/6	0/1	3/24

B.5 INDICATOR CATALOG

The catalog below summarizes the frozen rules. Task-level metadata-completeness measures use observed memberships as their denominator; semantic coverage uses applicable memberships with known values. Bin coverage counts occupied bins against all defined bins. Protocol rules operate on a benchmark protocol/version and retain confirmed, partial, unknown, or not-applicable states. Inference and conflict handling follow Section 3.2.

For scale, the six n bins have boundaries $0, 500, 2000, 10000, 100000, 1000000, \infty$; the six d bins use $0, 10, 50, 200, 1000, 10000, \infty$; the five n/d bins use $0, 1, 10, 100, 1000, \infty$. Intervals are left-closed and right-open. These audit bins differ from the coarser model-profile groups.

Table 13: The 44 indicator contracts. D/T rows aggregate memberships; D15–D16 summarize their distributions. E rows describe protocols.

ID	Indicator	Core rule
Identity and source diversity		
D01	Task identity coverage	Canonical identities / observed memberships.
Scale and geometry		
D02	Row count metadata coverage	Known row counts / memberships.
D03	Sample size bin coverage	Occupied sample-size bins / six bins.
D04	Feature count metadata coverage	Known feature counts / memberships.
D05	Feature count bin coverage	Occupied feature-count bins / six bins.
D06	N over d metadata coverage	Known positive-dimensional n/d / memberships.
D07	N over d bin coverage	Occupied n/d bins / five bins.
Feature structure and data quality		
D08	Feature type metadata coverage	At least one known feature-type count / memberships.
D09	Categorical task share	$n_{cat} > 0$ / tasks with known categorical counts.
D10	Datetime task share	$n_{time} > 0$ / tasks with known datetime counts.
D11	Text task share	$n_{text} > 0$ / tasks with known text counts.
D12	Missingness metadata coverage	Known missing-cell fractions / memberships.
D13	Any missing task share	Positive missing-cell fraction / tasks with known fractions.
D14	High cardinality task share	Maximum categorical cardinality $\geq$ 100 / applicable tasks with known cardinality.
Identity and source diversity		
D15	Source entropy	Normalized source entropy; retain category counts and unknown mass.
D16	Domain entropy	Normalized domain entropy; retain category counts and unknown mass.
Prediction-task diversity		
T01	Target identity coverage	Identified targets / memberships.
T02	Problem type metadata coverage	Normalized problem types / memberships.
T03	Classification task share	Classification / known classification-or-regression tasks.
T04	Regression task share	Regression / known classification-or-regression tasks.
T05	Binary classification share	Binary / classification tasks with known subtype.
T06	Multiclass classification share	Multiclass / classification tasks with known subtype.
T07	Class count metadata coverage	Known class counts / classification tasks.
T08	Many class task share	At least ten classes / classification tasks with known counts.
Generalization and replication		
T09	Split metadata coverage	Known split regimes / memberships.
T10	Iid task share	IID / interpretable split regimes.
T11	Temporal task share	Temporal / interpretable split regimes.
T12	Group or domain ood task share	Group or domain OOD / interpretable split regimes.
T13	Feature or schema shift task share	Feature or schema shift / interpretable split regimes.
T14	Replication metadata coverage	Both fold and repeat counts reported or explicitly inapplicable / memberships.
Comparison fairness		
E01	Same split parity disclosed	Evidence that models use the same splits.
E02	Preprocessing policy disclosed	Disclosure of preprocessing policy.
E03	Information parity disclosed	Disclosure of model information-access parity.

ID	Indicator	Core rule
E04	Hpo budget disclosed	Disclosure of hyperparameter-search budget.
E05	Compute budget disclosed	Disclosure of compute budget.
Evaluation coverage		
E06	Primary metric suitability documented	Evidence supporting primary-metric suitability.
E07	Calibration metric present	Presence of a calibration metric.
E08	Robustness metric present	Presence of a robustness metric.
E09	Efficiency and resource metrics present	Presence of efficiency/resource metrics.
Reliability and reproducibility		
E10	Failed run policy disclosed	Disclosure of failed-run handling.
E11	Aggregation and weighting disclosed	Disclosure of aggregation and weighting.
E12	Statistical uncertainty reported	Reported statistical uncertainty.
E13	Raw results or predictions available	Availability of raw results or predictions.
E14	Version and environment disclosed	Disclosure of versions and environment.

C Candidate Comparison and Benchmark Composition

C.1 From recovered inventory to executable candidates

The inventory contains 1,143 source-task memberships, while the executable pool comprises 88 distinct base identities: 66 IID and 22 natural-shift tasks. This pool combines 80 validated configurations with eight additional TabArena candidates whose access, split, and domain evidence were recovered. Constructed perturbations form the separate Stress-Grid track.

Table 14 links nine excluded configurations from five underlying tasks to their recovered conditions and eligibility decisions. Two configurations of `internet_usage` were also held because the nominal target `Actual_Time` contained occupation categories, requiring a decision on sensitive-target use. These held and excluded tasks remain outside the final pool. Conversely, recovering data access and official splits enabled the 11 BeyondArena tasks in Table 15 to be included, adding three temporal and eight grouped tasks to the final benchmark.

Table 14: Eligibility decisions: all nine excluded configurations remain outside the candidate pool.

Task (configurations)	Recovered condition	Recorded decision
Broadway (3)	OpenML 452 identifies target `Count` but leaves its meaning unresolved.	Exclude: target semantics.
Christine (1)	OpenML 41142 describes an anonymized challenge without recoverable application identity.	Exclude from real-domain coverage.
Libras (2)	OpenML 42855 uses L15 from an original multiple-label/ranking representation.	Exclude single-target conversion.
Robert (1)	OpenML 41165 hides application identity; description and class metadata disagree.	Exclude: identity/objective ambiguity.
Chscase-foot (2)	OpenML 703 does not define the meaning of target `col_6`.	Exclude: target semantics.

C.2 Candidate generation and comparison definitions

All schemes draw from the same 88-task pool and return 80 tasks. IID-priority and low-cost selection first retain the least-cost task from each of the temporal, group, and domain-OOD families. IID-priority then orders candidates by IID membership, cost, and identifier; low-cost selection orders by cost and identifier. Source- and domain-balanced schemes retain all 22 natural-shift tasks, then greedily minimize the corresponding maximum share, HHI, cost, and identifier in that order.

Table 15: Natural-shift tasks admitted after data-access and official split recovery. Each task is retained in the final 80-task manifest.

Task	Prediction target	Split
Hotel booking demand	`IsCanceled`	Temporal
Garments worker productivity	`actual_productivity`	Temporal
SF permit time	`DaysToIssue`	Temporal
Mice protein trisomy discriminant	`class`	Grouped
Parkinsons biomedical voice measurements	`status`	Grouped
Early learning predictors	`child_total_elom`	Grouped
Telemonitoring Parkinsons biomedical voice measurements	`total_UPDRS`	Grouped
SAT11-HAND algorithm runtime	`runtime`	Grouped
ASP-POTASSCO classification	`algorithm`	Grouped
Video transcoding time prediction	`log_transcoding_time`	Grouped
Video game FPS prediction	`log_fps_ratio`	Grouped

Shift-enhanced and boundary-coverage schemes start from all natural-shift tasks. At each step, boundary-greedy selection lexicographically maximizes newly occupied split, sample-size, feature-count, and n/d bins, followed by domain and problem-type coverage; ties prefer lower source HHI, domain HHI, cost, and identifier. The policy-based scheme applies the same rule starting from the 56 non-TabArena candidates. The existing and preliminary collections supply two fixed reference candidates. Sensitivity reporting deduplicates the nine schemes into seven task collections.

For a collection S, the natural-shift shortfall is

$$D_{\text{shift}}(S) = \max(0, 5 - N_{\text{temporal}}) + \max(0, 13 - N_{\text{group}}) + \max(0, 4 - N_{\text{domain}}).$$

The quotas retain the pool's five temporal, 13 grouped, and four domain-OOD tasks; shortfall counts unfilled slots against this coverage target. The metadata cost proxy is

$$C(S) = \sum_{t \in S} \log(1 + \max(1, n_t) \max(1, d_t)),$$

and source/domain concentration is $\text{HHI} = \sum_k p_k^2$, with p_k the corresponding task share. Duplicate count is $|S|$ minus the number of distinct base identities. License-record shortfall counts candidates without an explicit license entry, excluding generic entries such as *public* or *unknown*.

Pareto comparison minimizes shift shortfall, duplicates, excess source/domain shares, source/domain HHI, unknown-domain count, license-record shortfall, and cost; it maximizes occupied n, d, and n/d bins. A collection dominates another if it is no worse on every objective and strictly better on at least one (numerical tolerance 10^{-12}). The priority order in Appendix A.2 chooses among non-dominated candidates. All 44 indicators describe each collection; the policy selects this explicit subset as decision objectives rather than summing the profile into a quality score.

Tables 16 and 17 report candidate trade-offs and the final track composition.

Table 16: Score-blind comparison at 80 tasks per collection. Deficit counts unmet shift requirements; HHI measures domain concentration; cost is the metadata-based computation proxy. The selected set is also returned by the shift-enhanced and boundary-coverage schemes.

Collection	Shift deficit	Max. source	Max. domain	Domain HHI	Cost
Existing (v0.3)	0	30.0%	18.8%	0.114	1002.5
Preliminary candidate	0	30.0%	16.2%	0.106	990.1
IID priority	8	40.0%	17.5%	0.113	956.2
Source balanced	0	30.0%	20.0%	0.113	974.5
Domain balanced	0	33.8%	16.2%	0.106	978.4
Low cost	6	38.8%	18.8%	0.118	954.7
Selected (v0.4)	0	30.0%	16.2%	0.106	989.1

Table 17: AutoTabBench v0.4: 80 base tasks plus separately reported Controlled Stress.

Track	Condition	Count	Evaluation purpose
Real-Core	Routine prediction	58 tasks	In-distribution performance
Natural Shift	Group	13 tasks	Held-out groups
	Temporal	5 tasks	Later observations
	Domain OOD	4 tasks	Held-out domains
Controlled Stress	Perturbations	8 families	Controlled sensitivity

D Audit Corpus and Source Attribution

Table 18 maps the 30 source records to the 37 benchmark entries in the audit registry. Eight formation sources contribute 15 entries; 22 external-validation sources contribute one entry each. A source may be a model paper, benchmark paper, or official software resource. Registry identifiers distinguish the evaluated collections within a source; they are not necessarily names assigned by its authors. The audit corpus includes text, multimodal, generation, and multitask evaluations, whereas AutoTabBench selects predictive classification and regression tasks under the policy in Section 5.

Table 18: Sources and benchmark entries. Identifiers follow the frozen audit registry. F: indicator formation; E: external validation.

Source	Entries	Registry identifiers
F: Formation sources (8 sources; 15 entries)		
TabPFN (Hollmann et al., 2025)	3	`pfn_cls`, `pfn_reg`, `kaggle_tps_subset`
TALENT (Liu et al., 2025)	2	`talent_basic`, `talent_large`
LimiX/BCCO (Zhang et al., 2025)	4	`bcco_cls`, `bcco_reg`, `limix_imputation`, `limix_generation`
TabArena (Erickson et al., 2026)	1	`tabarena_v01`
TabZilla (McElfresh et al., 2023)	2	`tabzilla_full`, `tabzilla_hard`
OpenML-CC18 (Bischl et al., 2021)	1	`openml_cc18`
OpenML-CTR23 (Fischer et al., 2023)	1	`openml_ctr23`
WhyTrees (Grinsztajn et al., 2022)	1	`whytrees`
E: External-validation sources (22 sources; 22 entries)		
TableShift (Gardner et al., 2023)	1	`tableshift`
TabReD (Rubachev et al., 2025)	1	`tabred`
TabFSBench (Cheng et al., 2025)	1	`tabfsbench`
BeyondArena (Purucker et al., 2026)	1	`beyondarena`
AutoML Benchmark (Gijsbers et al., 2024)	1	`amlb`
TabRepo (Salinas & Erickson, 2024)	1	`tabrepo`
PMLBmini (Knauer et al., 2024)	1	`pmlbmini`
PMLB-v1 (Romano et al., 2022)	1	`pmlb_v1`
RTDL (Gorishniy et al., 2021)	1	`rtdl_benchmark`
TP-BERTa (Yan et al., 2024)	1	`tpberta_downstream`
TFM-OOD9 (Loza et al., 2026)	1	`tfm_ood9`
TFM-CDE (Izbicki & Rodrigues, 2026)	1	`tfm_cde`
TALENT-Uncertainty (De Melo Costa et al., 2026)	1	`talent_uncertainty`
TabSTAR-Text (Arazi et al., 2025)	1	`tabstar_text`
AutoML-Text-Tabular (Shi et al., 2021)	1	`automl_text_tabular`
AutoGluon-Multimodal (Tang et al., 2024)	1	`autogluon_multimodal`
MulTaBench (Arazi et al., 2026)	1	`multatab_bench`
TABLET (Slack & Singh, 2023)	1	`tablet`
TabuLa-8B (Gardner et al., 2024)	1	`tabula8b`
SDGym (DataCebo)	1	`sdgym`
TabularFM-Generation (Tran et al., 2024)	1	`tabularfm_generation`
MultiTab-Bench (Sinodinos et al., 2026)	1	`multitab_bench`

E Cross-Benchmark Audit Summary

Table 19 summarizes the 37-entry corpus. Sixteen entries have expanded task inventories totaling 1,143 memberships; the other 21 have benchmark-level audit cards, with unexpanded task inventories marked by a dash.

The D/T/E columns count computable indicators; U and NA count unknown and not-applicable indicator states. The final column counts unknown/conflict fields in audit cards, a separate unit. Registry identifiers link rows to Table 18 and evidence records.

Table 19: Structured audit coverage. D/T/E: known indicator counts out of 16/14/14; U/NA: indicator states; Card U/C: unknown/conflict card fields.

Entry	Tasks	Recovered split	D/T/E	U	NA	Card U/C
tabarena_v01	51	Cross-validation	15/14/8	7	0	0/0
beyondarena	142	Random, group, temporal	16/14/3	11	0	0/0
talent_basic	300	Fixed random holdout	15/14/3	11	1	12/0
talent_large	22	Fixed random holdout	15/14/3	12	0	12/0
bcco_cls	106	Provided / holdout	13/12/4	11	4	4/0
bcco_reg	50	Provided / holdout	13/10/4	10	7	4/0
tabzilla_full	177	Cross-validation	15/14/3	12	0	0/2
tabzilla_hard	36	Cross-validation	15/14/3	12	0	0/2
openml_cc18	72	Cross-validation	15/14/1	14	0	1/1
openml_ctr23	35	Cross-validation	15/10/4	10	5	1/2
pfn_cls	29	Repeated random	15/14/6	9	0	0/0
pfn_reg	28	Repeated random	15/10/5	9	5	0/0
amlb	–	–	0/0/5	39	0	1/0
tabrepo	–	–	0/0/7	37	0	0/1
whytrees	59	Cross-validation	15/14/4	11	0	1/1
tableshift	16	Domain ID/OOD	14/14/3	12	1	0/0
tabred	8	Temporal + alternatives	15/14/2	13	0	1/4
pmlbmini	–	–	0/0/5	39	0	0/0
tabfsbench	12	Feature shift	13/14/3	12	2	1/5
pmlb_v1	–	–	0/0/9	35	0	2/0
tpberta_downstream	–	–	0/0/2	42	0	2/0
rtdl_benchmark	–	–	0/0/3	41	0	4/0
kaggle_tps_subset	–	–	0/0/6	38	0	0/0
tfm_ood9	–	–	0/0/3	41	0	1/2
tfm_cde	–	–	0/0/3	40	1	2/4
talent_uncertainty	–	–	0/0/4	40	0	5/2
tabstar_text	–	–	0/0/2	42	0	1/1
automl_text_tabular	–	–	0/0/4	40	0	1/0
autogluon_multimodal	–	–	0/0/4	40	0	1/0
multatab_bench	–	–	0/0/3	41	0	0/2
tablet	–	–	0/0/3	41	0	2/0
tabula8b	–	–	0/0/4	40	0	1/1
limix_imputation	–	–	0/0/4	40	0	4/0
limix_generation	–	–	0/0/4	40	0	4/0
sdgym	–	–	0/0/5	39	0	0/1
tabularfm_generation	–	–	0/0/4	40	0	0/1
multitab_bench	–	–	0/0/4	40	0	1/1

Unknown counts aggregate unresolved states across audit depths. Appendix A.1 distinguishes insufficient source evidence from task-level examination awaiting expansion.

The expanded inventory includes 18 grouped and 21 temporal memberships in BeyondArena, 16 domain-shift memberships in TableShift, eight temporal-default memberships in TabReD, and 12 feature-shift memberships in TabFSBench. These counts describe recovered evaluation conditions

and may include repeated base datasets across benchmarks. Section 4.1 interprets selected protocol findings at their source locations; unresolved fields remain evidence gaps.

F Additional Model Results and Analysis Details

Evaluated configurations and preprocessing. The foundation-model configurations are TabPFN v3, TabICLv2, and LimiX 16M without retrieval. All seven models use the frozen default/zero-shot track with no task-specific hyperparameter search. Shared numerical preprocessing fits transformations on training data, imputes numerical missing values with training medians, and encodes missing or unseen categories as -1. Stress transformations act on raw data before this pipeline. Classification uses log-loss and regression uses RMSE.

F.1 Model execution settings

The evaluated server environment uses Python 3.11.16, PyTorch 2.5.1, NumPy 1.26.4, and NVIDIA RTX 3090 GPUs with 24 GB memory. Foundation-model execution uses CUDA; tree estimators use one CPU thread per cell. The per-cell timeout is 3,600 seconds and seeds are 0, 1, and 2. No guard silently subsamples a task to obtain a successful run.

Model	Implementation / assets	Inference settings and guards
TabPFN	Package 8.0.3; v3 default classifier/regressor checkpoints	Seeded default estimator; at most 10^6 training rows, 2,000 features, and 160 classes. Prediction batching can reduce memory use without dropping test rows.
TabICL	Package 2.1.1; classifier/regressor v2 checkpoints dated 20260212	Seeded default estimator; batch size 8 with automatic offload; memory-recovery configuration uses batch size 1 and disk offload.
LimiX	Official source; LimiX-16M checkpoint	No retrieval; task-specific default no-retrieval configurations. Classification requires fewer than 50,000 training rows and at most ten classes. Regression targets are standardized from training mean/standard deviation and predictions inverse-transformed.
CatBoost	1.2.10	Library defaults, fixed seed, one thread, silent logging.
XGBoost	2.1.4	Library defaults, histogram tree method, fixed seed, one thread.
LightGBM	4.6.0	Library defaults, fixed seed, one thread, silent logging.
ExtraTrees	scikit-learn 1.6.1	500 trees, fixed seed, one thread.

TabPFN uses the v3 checkpoints (Grinsztajn et al., 2026) with package version 8.0.3 from the pre-experiment runtime archive dated August 21, 2026. Memory-control changes leave task selection, model checkpoints, evaluation metrics, and random seeds unchanged.

All models receive shared floating-point arrays with categorical columns ordinal-encoded using training-fitted vocabularies; missing and unseen categories receive -1. CatBoost, LightGBM, and XGBoost use these as numerical inputs, with their native categorical interfaces disabled. Foundation models use package defaults and the same arrays without categorical-column metadata.

F.2 Exact ranking convention

For task t, own-success loss averages the available successful seeds of model m. Available models receive ascending loss ranks with average ranks for ties; each model's own-success aggregate averages over its available tasks. For failure-last ranking, let k_t be the number of models completing all three seeds. These models are ranked by their three-seed mean losses; each remaining model receives

$$r_{tm}^{\mathrm{FL}} = \frac{(k_t + 1) + 7}{2}.$$

For example, with five complete models, both incomplete models receive rank 6.5. If no model completes, all receive rank 4. Task-equal results average these ranks over the complete task collection; common-success analyses require all seven models to complete all three seeds.

F.3 CONTROLLED TRANSFORMATIONS

Transformations act on raw data before the shared train-fitted preprocessing pipeline. Test-only stresses preserve the training set and its fitted preprocessing. Training-label or row-selection stresses alter the corresponding training inputs before fitting. Random selections and Gaussian draws use the recorded stress seed. The following definitions describe the settings displayed in the main figures.

Stress	Scope	Transformation
Minority retention 0.1	Train	Keep the largest class and sample 10% of each other class, retaining at least one example per class. Classification only.
Feature masking 0.5	Test	Rank columns by decreasing training-set nonmissing count, breaking ties by column index; mask the first half of the columns in every test row.
Missingness 0.4	Test	Uniformly shuffle observed test cells and hide 40% of them; already missing cells are not sampled.
Noise $1\times$IQR	Test	Add independent zero-mean Gaussian noise to observed numerical entries, with standard deviation $\max(\mathrm{IQR}_{j,\mathrm{train}}, 10^{-12})$ for feature j.
Expansion $2\times$	Test	Map numerical entries to $m_j + 2(x_j - m_j)$, where m_j is the training median.
Unseen category 0.5	Test	Sample half of rows having an observed categorical entry; replace one randomly selected eligible entry per sampled row with an unseen token.
Label noise 0.2	Train	Sample 20% of training labels. For classification, replace with a uniformly selected different observed class. For regression, add Gaussian noise with standard deviation $0.2\max(\mathrm{IQR}_{y,\mathrm{train}}, 10^{-12})$.
Sample budget	Train	Subsample the original training partition to 5%, 10%, 25%, and 50%, keeping validation and test partitions unchanged. Results are reported in Table 23.

Fraction-based selections round to the nearest integer with a minimum of one selected item. Numerical stresses use the registered numerical columns; categorical stresses use registered categorical columns. Missing and unseen categories pass through the shared imputation/encoding pipeline, so the measured response includes that preprocessing.

F.4 CATEGORICAL-INTERFACE SENSITIVITY

We compare shared numerical encoding with native categorical interfaces on 41 tasks with categorical columns in the evaluated inputs, holding splits, seeds, numerical preprocessing, and model settings fixed. CatBoost receives string-valued categories; LightGBM and XGBoost receive training-vocabulary categorical columns, with out-of-vocabulary values mapped to missing. Stress comparisons use applicable Real-Core tasks under numerical noise ($1\times$ training IQR), added missingness (0.4), and unseen categories (0.5).

Table 20: Native/shared loss ratios across paired tasks, using three-seed mean losses per task. Ratios below 1 favor native interfaces. W/T/L counts native wins/ties/losses.

Condition	Model	Tasks	Median	Geom. mean	W/T/L
Unperturbed	CatBoost	39	0.997	1.011	23/0/16
	LightGBM	41	1.000	0.971	19/4/18
	XGBoost	41	1.000	0.972	17/4/20
Numerical noise	CatBoost	26	0.985	0.968	19/0/7
	LightGBM	26	1.000	0.999	12/2/12
	XGBoost	26	0.989	0.880	16/2/8
Missingness	CatBoost	30	0.963	0.888	20/0/10
	LightGBM	30	1.004	1.010	13/1/16
	XGBoost	30	1.064	1.187	7/0/23
Unseen categories	CatBoost	30	0.990	0.945	18/0/12
	LightGBM	30	1.001	0.987	13/1/16
	XGBoost	30	1.035	1.111	8/0/22

Unperturbed median loss ratios are close to one, while stress responses vary by model (Table 20). Native CatBoost improves on 20 of 30 missingness tasks; native XGBoost has higher loss on 23 of 30 missingness tasks and 22 of 30 unseen-category tasks. Native interfaces therefore offer no uniform advantage in this comparison. Of 2,286 applicable runs, 2,280 succeed; native CatBoost reaches the one-hour limit on all three seeds of both lending_club and consumer_complaints. These two tasks are excluded from its loss pairing. On the 39 paired unperturbed tasks, native CatBoost takes a median 3.65 times as long as shared encoding.

F.5 Common-success results

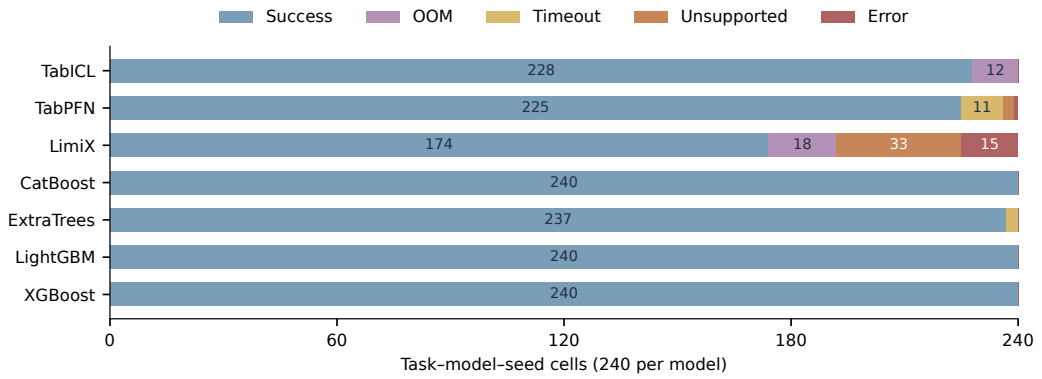

Figure 5: Terminal outcomes across 240 base cells per model, including failures. Counts: Table 21.

Common-success comparisons require all three seeds of all seven models to succeed. The resulting subsets contain 49 Real-Core tasks and nine Natural Shift tasks. Table 21 reports their ranks together with the complete base-track failure counts. Timeout is the recorded 3,600-second limit. Unsupported outcomes reflect the frozen execution guards: LimiX has 24 sample-cap and nine class-cap exclusions; TabPFN has three feature-cap exclusions. These reflect model settings and limits.

Table 21: Common-success ranks (lower is better) and non-success cells across 1,680 base cells.

	Common-success rank		Non-success cells			
Model	Core (49)	Shift (9)	OOM	Timeout	Unsupported	Error
TabICL	1.980	2.889	12	0	0	0
TabPFN	2.082	4.111	0	11	3	1
LimiX	3.082	2.889	18	0	33	15
CatBoost	3.980	3.444	0	0	0	0
LightGBM	5.429	5.667	0	0	0	0
ExtraTrees	5.571	3.889	0	3	0	0
XGBoost	5.878	5.111	0	0	0	0

TabICL's OOM records comprise seven CUDA allocation failures, four CPU-allocation/offload failures, and one signal-9 termination of undetermined cause recorded as OOM. TabPFN's error is a signal-11 worker exit without further diagnostic output. LimiX's 15 errors report an invalid CUDA configuration, with the underlying cause unresolved in the asynchronous stack traces.

Descriptive uncertainty uses 2,000 task-level bootstrap samples with percentile intervals, averaging seeds within tasks before resampling. These intervals describe variation within the evaluated collection, whose tasks can share sources.

Real-Core · tasks 1–29

Task / dataset	Loss	TabICL	TabPFN	LimiX	CatBoost	ExtraTrees	LightGBM	XGBoost
abalone	RMSE	**2.030**	2.053	2.068	2.200	2.207	2.185	2.330
airfoil self noise	RMSE	**0.919**	0.929	1.057	1.350	1.220	1.667	1.618
Airlines DepDelay 1M	RMSE	NC (0/3)	NC (0/3)	NC (0/3)	**1.908**	2.014	1.911	1.908
Amazon employee access	Log loss	**0.165**	0.165	0.165	0.185	0.302	0.184	0.187
anneal	Log loss	0.057	0.048	**0.039**	0.059	0.091	0.156	0.060
Another-Dataset-on-used-Fiat-500	RMSE	687.358	**684.158**	703.890	731.923	740.954	777.631	837.120
APSFailure	Log loss	**0.016**	0.017	NC (0/3)	0.019	0.027	0.022	0.020
auction verification	RMSE	610.586	594.709	1.08e+03	603.230	1.36e+03	1.13e+03	**358.383**
blood-transfusion-service-center	Log loss	**0.475**	0.484	0.477	0.503	2.109	0.560	0.655
churn	Log loss	0.114	**0.103**	0.146	0.143	0.207	0.155	0.154
clock protein toxicity	Log loss	0.905	**0.702**	0.774	0.777	0.807	1.084	1.140
concrete compressive strength	RMSE	**5.376**	5.419	5.448	5.535	5.830	6.045	5.670
credit-g	Log loss	0.499	0.511	**0.484**	0.506	0.520	0.581	0.679
customer satisfaction in airline	Log loss	0.085	**0.084**	NC (0/3)	0.099	0.128	0.118	0.104
default-of-credit-card-clients	Log loss	0.553	0.554	**0.553**	0.560	0.688	0.564	0.600
diamonds	RMSE	513.804	**500.897**	553.254	541.838	557.383	559.650	563.441
E-CommereShippingData	Log loss	0.495	0.496	**0.495**	0.517	0.528	0.521	0.555
energy efficiency	RMSE	0.519	0.516	0.589	**0.448**	0.586	0.705	0.452
fifa	RMSE	8.95e+03	9.09e+03	**8.78e+03**	9.27e+03	9.74e+03	9.10e+03	1.05e+04
forest fires	RMSE	152.347	153.080	152.722	150.451	151.184	**147.213**	153.955
fps benchmark	RMSE	**0.343**	0.408	4.296	1.165	9.110	3.359	3.501
geographical origin of music	RMSE	13.807	**13.688**	13.859	15.487	15.235	16.596	16.451
grid stability	RMSE	**0.005**	0.005	0.006	0.007	0.011	0.009	0.010
hazelnut-spread-contaminant-detection	Log loss	0.116	**0.111**	0.158	0.256	0.302	0.280	0.258
health insurance	RMSE	**14.700**	14.735	14.712	14.822	16.625	14.771	15.304
house sales	RMSE	0.174	**0.168**	0.170	0.176	0.191	0.182	0.187
HR Analytics Job Change of Data Scientists	Log loss	0.423	**0.423**	0.424	0.435	0.565	0.430	0.466
in vehicle coupon recommendation	Log loss	0.492	**0.488**	0.532	0.513	0.622	0.516	0.545
Is-this-a-good-customer	Log loss	**0.319**	0.321	0.325	0.352	0.357	0.538	0.554

Numbers: mean loss over three seeds (lower is better). Blue shading and bars: within-task rank (darker / longer is better).
NC: not all three seeds completed; parentheses show successful seeds. Rankings include complete models only.

Figure 6: Task-level model losses on Real-Core (1/2). Cells report three-seed mean loss; shading and bars indicate within-task rank. NC marks incomplete runs.

F.6 Paired stress results and transformation scope

Sensitivity also varies across the frozen transformations. On their condition-specific common task sets, retaining 10% of non-majority-class examples yields median loss ratios of 1.799–3.030, com-

Real-Core · tasks 30–58

Task / dataset	Loss	TabICL	TabPFN	LimiX	CatBoost	ExtraTrees	LightGBM	XGBoost
jm1	Log loss	**0.415**	0.422	0.424	0.432	0.786	0.439	0.478
kings county	RMSE	9.64e+04	**9.33e+04**	9.45e+04	1.01e+05	1.16e+05	1.06e+05	1.13e+05
ljubljana primary tumor	Log loss	1.231	**1.197**	NC (0/3)	1.595	5.264	1.584	1.701
lung cancer epithelial genexp	Log loss	0.656	NC (0/3)	NC (0/3)	**0.576**	0.600	0.839	0.808
maternal health risk	Log loss	0.401	**0.397**	0.437	0.467	1.296	0.479	0.498
medical charges	RMSE	**0.080**	0.080	NC (0/3)	0.082	0.087	0.082	0.082
miami housing	RMSE	**7.95e+04**	8.09e+04	8.04e+04	8.45e+04	9.57e+04	9.22e+04	9.70e+04
MiniBooNE	Log loss	0.128	**0.122**	NC (0/3)	0.144	0.188	0.154	0.156
NATICUSdroid	Log loss	**0.139**	0.140	0.145	0.149	0.579	0.155	0.163
naval propulsion plant	RMSE	**1.02e-04**	2.16e-04	7.52e-04	0.001	5.84e-04	0.001	0.001
online shoppers intention	Log loss	0.226	**0.225**	0.229	0.245	0.264	0.248	0.286
physiochemical protein	RMSE	2.904	**2.879**	3.113	3.853	3.419	4.095	3.872
polish companies bankruptcy	Log loss	0.119	**0.074**	0.090	0.111	0.196	0.137	0.132
qsar-biodeg	Log loss	**0.304**	0.309	0.308	0.312	0.326	0.399	0.423
QSAR fish toxicity	RMSE	0.946	0.936	0.945	**0.925**	0.949	0.987	1.033
red wine	RMSE	0.604	**0.591**	0.610	0.627	0.604	0.639	0.655
seattlecrime6	RMSE	**373.752**	374.226	379.219	374.156	374.524	373.874	374.510
seismic-bumps	Log loss	**0.204**	0.205	0.206	0.214	0.313	0.320	0.281
SGEMM GPU kernel performance	RMSE	0.015	0.015	NC (0/3)	0.017	**0.014**	0.017	0.017
solar flare	RMSE	**1.069**	1.093	1.109	1.072	1.093	1.069	1.069
splice	Log loss	**0.074**	0.079	0.080	0.112	0.367	0.137	0.119
student performance por	RMSE	2.140	2.117	**2.097**	2.326	2.527	2.595	2.732
students dropout and academic success	Log loss	**0.558**	0.567	0.561	0.631	0.638	0.663	0.737
superconduct	RMSE	**9.043**	9.160	9.703	10.076	9.781	10.645	10.211
superconductivity	RMSE	**9.073**	9.173	9.680	10.087	9.773	10.702	10.114
video transcoding	RMSE	0.854	**0.619**	2.883	1.781	1.743	1.773	1.751
wave energy	RMSE	4.72e+03	**4.49e+03**	NC (0/3)	5.64e+03	4.35e+04	2.72e+04	2.30e+04
white wine	RMSE	0.544	0.543	**0.536**	0.596	0.547	0.607	0.599
wine quality	RMSE	0.602	**0.598**	0.599	0.640	0.611	0.649	0.663

Numbers: mean loss over three seeds (lower is better). Blue shading and bars: within-task rank (darker / longer is better).
NC: not all three seeds completed; parentheses show successful seeds. Rankings include complete models only.

Figure 7: Task-level model losses on Real-Core (2/2), using the same encoding as Figure 6.

Natural Shift · 22 tasks

Task / dataset	Loss	TabICL	TabPFN	LimiX	CatBoost	ExtraTrees	LightGBM	XGBoost
brfss blood pressure [OOD]	Log loss	0.682	0.680	NC (0/3)	0.677	2.015	**0.676**	0.679
brfss diabetes [OOD]	Log loss	NC (0/3)	NC (0/3)	NC (0/3)	0.371	NC (0/3)	**0.369**	0.370
diabetes readmission [OOD]	Log loss	0.640	**0.637**	NC (0/3)	0.638	0.663	0.639	0.654
nhanes lead [OOD]	Log loss	**0.257**	0.277	0.273	0.270	0.514	0.281	0.275
5g energy consumption [group]	RMSE	3.989	**3.966**	NC (0/3)	4.009	4.512	3.991	4.059
asp potassco classification [group]	Log loss	**2.182**	2.225	NC (0/3)	2.432	3.214	4.235	3.152
cardiotocography [group]	Log loss	**0.307**	0.318	0.324	0.349	0.350	0.605	0.442
covertype [group]	Log loss	0.950	0.830	NC (0/3)	0.883	**0.714**	0.800	1.002
early learning predictors [group]	RMSE	13.150	**12.917**	NC (0/3)	13.047	13.351	13.159	13.565
mice protein trisomy discriminant [group]	Log loss	0.985	1.101	**0.934**	1.191	1.253	2.435	1.824
micro mass [group]	Log loss	**0.401**	0.639	NC (0/3)	0.578	0.861	0.792	0.882
musk [group]	Log loss	2.023	1.915	0.823	0.766	0.581	0.794	**0.440**
parkinsons biomedical voice measurements [group]	Log loss	0.422	0.493	0.363	**0.322**	0.335	0.407	0.375
sat11 hand algo runtime [group]	RMSE	2.039	1.969	**1.880**	1.880	1.892	2.132	2.100
telemonitoring parkinsons biomedical voice measurements [group]	RMSE	**9.932**	10.673	10.903	10.969	10.502	11.006	12.094
video game fps prediction [group]	RMSE	0.007	0.007	**0.005**	0.007	0.023	0.017	0.023
video transcoding time prediction [group]	RMSE	0.272	0.265	NC (0/3)	0.264	**0.249**	0.250	0.286
consumer complaints 1m [time]	Log loss	NC (0/3)	NC (0/3)	NC (0/3)	0.481	0.504	**0.478**	0.490
garments worker productivity [time]	RMSE	**0.073**	0.080	0.088	0.092	0.079	0.085	0.089
hotel booking demand [time]	Log loss	1.845	3.117	NC (0/3)	1.154	1.086	**0.664**	0.827
lending club 1m [time]	Log loss	NC (0/3)	NC (0/3)	NC (0/3)	0.504	0.509	**0.498**	0.503
sf permit time [time]	RMSE	1.565	1.496	NC (0/3)	1.545	**1.483**	1.574	1.641

Numbers: mean loss over three seeds (lower is better). Blue shading and bars: within-task rank (darker / longer is better).
NC: not all three seeds completed; parentheses show successful seeds. Rankings include complete models only.

Figure 8: Task-level model losses on Natural Shift, using the same encoding as Figure 6.

pared with 1.019–1.040 for replacement of one categorical value in 50% of applicable test rows. These outcomes include the shared preprocessing pipeline's response to the perturbations.

Table 3 in the main text reports the fixed 46-task numerical-noise comparison. For each task and model, unperturbed and stressed losses are separately averaged over the three paired seeds; their ratio is then computed before taking the median across tasks. Unperturbed and stressed ranks are computed within each task across the same seven models, then averaged over the same 46 tasks.

Table 22 uses a condition-specific common-success subset. Ratios measure relative degradation, with 1 indicating unchanged loss; ratios exclude unperturbed losses at or below 10^{-12}.

Table 22: Median paired loss ratios at selected strong stress settings. CB: CatBoost; ET: ExtraTrees; LGB: LightGBM; XGB: XGBoost. N is the number of jointly successful tasks.

Condition	N	CB	ET	LGB	XGB	TabICL	TabPFN	LimiX
Minority retention 0.1	21	2.128	2.180	3.030	2.395	1.907	1.964	1.799
Feature masking 0.5	49	1.663	1.384	1.640	1.703	1.530	1.623	1.467
Added missingness 0.4	49	1.620	1.318	1.551	1.616	1.651	1.684	1.571
Numerical noise 1×IQR	46	1.631	1.548	1.753	1.829	1.707	2.100	1.917
Numerical expansion 2×	46	1.386	1.328	1.316	1.414	1.409	1.587	1.499
Unseen-category rate 0.5	27	1.020	1.019	1.020	1.037	1.040	1.030	1.027
Training-label noise 0.2	49	1.037	1.071	1.018	1.057	1.083	1.068	1.037

The stress definitions in Appendix F.3 specify each parameter's meaning. In particular, the regression label-noise parameter controls both the fraction of affected labels and the Gaussian noise scale.

Paired uncertainty under numerical noise. For the fixed 46-task comparison, define $\Delta_{AB} = (r_A^{\text{noise}} - r_B^{\text{noise}}) - (r_A^{\text{base}} - r_B^{\text{base}})$. Positive values indicate a relative rank change favoring model B. We average seeds within each task, rank the same seven models, and resample the 46 task IDs jointly across both conditions and all models. Table 4 reports 2,000 task-bootstrap replicates (seed 20260914), using the 50th and 1950th ordered bootstrap estimates for the descriptive 95% interval. Pairs follow the leaders highlighted in Table 3; intervals are post-hoc and pointwise. Shared data sources can induce dependence across tasks.

Training-sample reduction. Training-sample reduction separates predictive ranking from relative loss growth (Table 23). At the 5% setting, TabPFN has both the best mean rank (2.224) and the smallest median loss ratio (1.112). At 50%, TabPFN remains first by rank (1.959), whereas LimiX has the smallest loss ratio (1.027). Median loss ratios decrease with increasing budget for all seven models. Each setting uses 49 tasks with three successful unperturbed–stress seed pairs for every model.

Table 23: Training-sample reduction on 49 common-success tasks per setting. Cells show mean rank / median paired loss ratio; lower is better.

Training budget	5%	10%	25%	50%
TabICL	2.837 / 1.165	2.551 / 1.130	1.939 / 1.073	2.082 / 1.031
TabPFN	2.224 / 1.112	2.020 / 1.106	2.224 / 1.059	1.959 / 1.040
LimiX	2.633 / 1.130	2.694 / 1.111	2.939 / 1.051	3.082 / 1.027
CatBoost	3.857 / 1.143	4.000 / 1.140	4.082 / 1.059	4.082 / 1.040
LightGBM	5.490 / 1.211	5.327 / 1.178	5.490 / 1.101	5.551 / 1.054
ExtraTrees	5.061 / 1.124	5.245 / 1.086	5.347 / 1.048	5.531 / 1.032
XGBoost	5.898 / 1.300	6.163 / 1.251	5.980 / 1.131	5.714 / 1.076

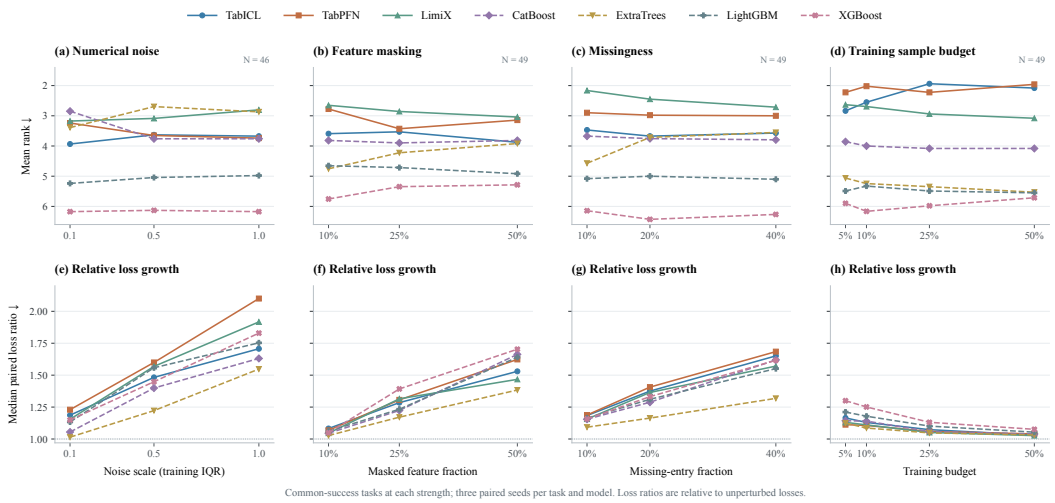

Figure 9: Mean ranks (top) and median paired loss ratios (bottom) across four stress families. Lower values are better; each column reports its common-success task count.

Stress responses across seven models

Loss reduction relative to unperturbed conditions (%)

Stress family	Setting	N	TabICL	TabPFN	LimiX	CatBoost	ExtraTrees	LightGBM	XGBoost
Minority-class retention	50% retained	21	-12.9	-13.2	-12.3	-13.6	-13.0	-25.9	-18.2
	25% retained	21	-41.2	-40.7	-36.8	-42.9	-34.7	-68.9	-54.8
	10% retained	21	-90.7	-96.4	-79.9	-112.8	-118.0	-203.0	-139.5
Feature masking	10% masked	49	-8.2	-6.8	-4.6	-4.5	-2.7	-6.1	-4.9
	25% masked	49	-28.5	-30.6	-31.5	-22.4	-17.1	-23.3	-39.2
	50% masked	49	-53.0	-62.3	-46.7	-66.3	-38.4	-64.0	-70.3
Added missingness	10% cells	49	-18.4	-18.7	-15.6	-15.5	-9.3	-16.4	-15.7
	20% cells	49	-37.5	-40.6	-36.5	-28.8	-16.4	-31.1	-33.2
	40% cells	49	-65.1	-68.4	-57.1	-62.0	-31.8	-55.1	-61.6
Numerical noise	0.1 × IQR	46	-18.7	-23.0	-15.5	-5.5	-1.4	-13.3	-14.9
	0.5 × IQR	46	-48.3	-59.9	-56.9	-39.9	-22.2	-55.8	-44.8
	1.0 × IQR	46	-70.7	-110.0	-91.7	-63.1	-54.8	-75.3	-82.9
Numerical scaling	0.5 × scale	46	-34.2	-54.6	-35.1	-24.9	-19.4	-29.0	-33.9
	1.5 × scale	46	-26.8	-31.1	-21.9	-16.8	-15.6	-16.9	-21.8
	2.0 × scale	46	-40.9	-58.7	-49.9	-38.6	-32.8	-31.6	-41.4
Unseen categories	10% rows	27	-0.6	-0.4	-0.7	-0.3	-0.2	-0.3	-0.2
	25% rows	27	-2.1	-1.3	-1.7	-1.1	-0.7	-1.0	-1.9
	50% rows	27	-4.0	-3.0	-2.7	-2.0	-1.9	-2.0	-3.7
Training-label noise	0.05	49	-0.1	-0.2	-0.04	-0.4	-0.2	-0.01	-0.5
	0.10	49	-1.7	-1.5	-0.4	-1.0	-0.7	-0.4	-2.6
	0.20	49	-8.3	-6.8	-3.7	-3.7	-7.1	-1.8	-5.7
Training-sample budget	5% retained	49	-16.5	-11.2	-13.0	-14.3	-12.4	-21.1	-30.0
	10% retained	49	-13.0	-10.6	-11.1	-14.0	-8.6	-17.8	-25.1
	25% retained	49	-7.3	-5.9	-5.1	-5.9	-4.8	-10.1	-13.1
	50% retained	49	-3.1	-4.0	-2.7	-4.0	-3.2	-5.4	-7.6

Cell value = 100 × (1 − median paired loss ratio). Red: loss increase; blue: loss reduction. All bars use the same linear scale.
N: tasks jointly completed by all seven models at that setting; each ratio uses three-seed mean losses before the task median.
Training-label noise changes the fraction of labels; for regression, its parameter also sets the noise scale.

Figure 10: Model responses across eight stress families and their strength settings. Signed bars show $100(1 - r)$, where r is the median paired loss ratio; negative values indicate increased loss.

F.7 Composition and reporting settings

The analysis pool comprises 88 tasks and 1,848 terminal cells. The 132 settings include task-, identity-cluster-, source-, domain-, and sample-weighted aggregation; source/domain and other group omissions; IID weights of 0%, 25%, 50%, 75%, and 100%; task counts of 20, 40, 60, and 80 with ten fixed subset seeds; and metadata-cost budgets of 25%, 50%, 75%, and 100%. The cost proxy is based on $\log(1 + nd)$, rather than measured GPU time. These overlapping settings were specified at the analysis stage for descriptive sensitivity comparisons, not independent trials.

Table 24: First-ranked models under representative reporting choices. Pooled views combine the two base tracks for sensitivity analysis; Stress is separate. Available-task and common-success results use their respective task subsets.

Setting	Failure-last	Available tasks	Common success
Task-equal, pooled	TabICL	TabICL	TabICL
Task-equal, Real-Core	TabICL	TabICL	TabICL
Task-equal, Natural Shift	CatBoost	LimiX	TabICL / LimiX
Source-equal, pooled	TabICL	TabPFN	TabICL
Domain-equal, pooled	TabICL	TabICL	TabICL
Sample-weighted, pooled	LightGBM	LightGBM	TabICL
Omit small tasks	TabPFN	TabPFN	TabICL

Across the formal alternatives, LimiX's success rate ranges from 71.25% to 81.25%, TabPFN's from 93.75% to 97.50%, and TabICL's from 95.00% to 98.75%. CatBoost, XGBoost, and LightGBM complete all cells; ExtraTrees ranges from 98.75% to 100%. These coverage differences remain visible alongside the stability of the failure-last leader.

F.8 Condition-group model profiles

Figure 3 organizes the seven models by condition group. Within a task, successful-seed losses are averaged and the available models are ranked. Each model's condition average includes its own tasks with at least one successful seed. Stress spokes instead use the median paired loss ratio, requiring three successful unperturbed–stress seed pairs per task. Table 26 gives values and denominators for all groups. Size groups: small ($n \leq 2{,}000$), medium ($2{,}000 < n \leq 10{,}000$), and large ($n > 10{,}000$).

Table 25: Paired-task counts by model for Figure 4, requiring three successful unperturbed–stress seed pairs. CB: CatBoost; ET: ExtraTrees; LGB: LightGBM; XGB: XGBoost.

Stress	ICL	PFN	LimiX	CB	ET	LGB	XGB
Minority retention 0.1	26	25	21	26	26	26	26
Feature masking 0.5	57	56	49	58	58	58	58
Missingness 0.4	57	56	49	58	58	58	58
Noise 1× IQR	53	52	46	54	54	54	54
Expansion 2×	53	52	46	54	54	54	54
Unseen category 0.5	30	30	27	30	30	30	30
Label noise 0.2	57	56	49	58	58	58	58

Low and high dimensionality use frozen input feature counts below and at least 100, respectively. Numerical, mixed, and categorical groups use numerical/categorical feature counts from source-task metadata; high cardinality requires at least 100 values in a categorical feature with known metadata. Missingness groups use 0%, $(0, 10\%]$, and above 10%. Class-balance spokes use the smallest class's share of dataset rows: at most 10%, $(10, 25\%]$, and above 25%. These shares are available for 36 of the 40 classification tasks; the four unknown cases are excluded from those spokes. Registry joins match benchmark/task identity, dataset name, target, and row count. Application-domain spokes retain all 15 registry labels. Groups may overlap, and the thresholds are descriptive analysis choices.

The stress spokes use numerical noise at one training IQR, added missingness 0.4, feature masking 0.5, unseen-category rate 0.5, training-label noise 0.2, minority retention 0.1, and numerical expansion 2×. Their task-specific transformations are described in Appendix F.3.

Table 26: Condition-group radar values and denominators. Entries are metric (successful tasks); N is the eligible task count. Groups (a–f) use mean available-task rank; (g) uses median paired loss ratio. A dash indicates no successful tasks.

Condition	N	ICL	PFN	LimiX	CB	ET	LGB	XGB
(a) Split protocol								
IID	58	1.96 (57)	2.02 (56)	3.08 (49)	3.84 (58)	5.45 (58)	5.24 (58)	5.60 (58)
Group	13	3.15 (13)	3.31 (13)	2.57 (7)	3.08 (13)	3.77 (13)	4.77 (13)	5.31 (13)
Temporal	5	3.33 (3)	3.67 (3)	5.00 (1)	3.80 (5)	2.80 (5)	2.40 (5)	3.80 (5)
Domain OOD	4	3.33 (3)	3.33 (3)	3.00 (1)	2.25 (4)	6.33 (3)	2.75 (4)	3.50 (4)
(b) Prediction task								
Binary	29	2.52 (27)	2.54 (26)	2.75 (20)	3.28 (29)	5.61 (28)	4.55 (29)	4.97 (29)
Multiclass	11	1.90 (10)	2.20 (10)	2.17 (6)	3.55 (11)	5.09 (11)	4.91 (11)	5.36 (11)
Regression	40	2.21 (39)	2.28 (39)	3.41 (32)	3.92 (40)	4.62 (40)	5.08 (40)	5.60 (40)
(c) Dataset size								
Small	25	2.28 (25)	2.67 (24)	3.10 (21)	3.52 (25)	4.48 (25)	5.44 (25)	5.76 (25)
Medium	15	2.47 (15)	2.20 (15)	2.87 (15)	3.80 (15)	5.60 (15)	5.87 (15)	5.20 (15)
Large	40	2.19 (36)	2.22 (36)	3.14 (22)	3.65 (40)	5.18 (39)	4.12 (40)	5.12 (40)
(d) Feature structure								
Low dim.	68	2.09 (65)	2.34 (65)	3.07 (54)	3.82 (68)	5.15 (68)	4.94 (68)	5.49 (68)
High dim.	12	3.36 (11)	2.50 (10)	2.75 (4)	2.58 (12)	4.36 (11)	4.42 (12)	4.50 (12)
Numerical	35	2.38 (34)	2.33 (33)	2.85 (27)	3.43 (35)	4.89 (35)	5.54 (35)	5.34 (35)
Mixed	40	2.29 (38)	2.45 (38)	3.25 (28)	3.80 (40)	5.03 (39)	4.40 (40)	5.38 (40)
Categorical	5	1.25 (4)	1.75 (4)	3.00 (3)	3.80 (5)	6.20 (5)	3.80 (5)	5.00 (5)
High card.	9	2.57 (7)	2.43 (7)	3.25 (4)	3.56 (9)	4.89 (9)	3.11 (9)	4.78 (9)
(e) Missingness and class balance								
No missing	62	2.25 (60)	2.39 (59)	3.10 (50)	3.69 (62)	5.03 (61)	5.05 (62)	5.48 (62)
Missing $\leq$10%	12	2.33 (12)	2.50 (12)	2.75 (8)	3.75 (12)	5.50 (12)	4.67 (12)	5.08 (12)
Missing $>$10%	6	2.50 (4)	1.50 (4)	– (0)	2.83 (6)	4.17 (6)	3.33 (6)	4.33 (6)
Minority $\leq$10%	12	2.00 (11)	2.09 (11)	2.17 (6)	3.50 (12)	5.25 (12)	4.92 (12)	5.08 (12)
Minority 10–25%	11	2.40 (10)	2.80 (10)	3.10 (10)	3.55 (11)	5.36 (11)	4.82 (11)	5.09 (11)
Minority $>$25%	13	2.38 (13)	2.25 (12)	2.33 (9)	3.38 (13)	5.54 (13)	4.85 (13)	5.54 (13)
(f) Application domain								
Health	13	2.33 (12)	2.91 (11)	2.86 (7)	3.15 (13)	5.25 (12)	4.15 (13)	4.85 (13)
Finance	13	2.73 (11)	2.36 (11)	2.78 (9)	3.31 (13)	5.31 (13)	4.38 (13)	4.92 (13)
Computing	10	2.20 (10)	2.50 (10)	3.50 (6)	3.80 (10)	4.70 (10)	4.70 (10)	5.20 (10)
Life sci.	9	2.67 (9)	2.56 (9)	2.75 (8)	3.44 (9)	4.78 (9)	5.56 (9)	5.78 (9)
Industry	8	1.71 (7)	2.00 (7)	4.20 (5)	3.50 (8)	4.75 (8)	5.25 (8)	4.62 (8)
Engineering	6	1.17 (6)	1.83 (6)	3.00 (5)	4.33 (6)	5.00 (6)	6.50 (6)	5.50 (6)
Environment	4	3.00 (4)	4.00 (4)	6.00 (2)	3.25 (4)	3.75 (4)	2.50 (4)	5.00 (4)
Agriculture	4	2.75 (4)	1.25 (4)	2.50 (4)	4.75 (4)	4.25 (4)	6.25 (4)	6.25 (4)
Education	3	2.33 (3)	2.00 (3)	1.50 (2)	3.33 (3)	5.00 (3)	5.33 (3)	6.67 (3)
Housing	3	2.33 (3)	1.67 (3)	2.00 (3)	4.00 (3)	6.67 (3)	5.00 (3)	6.33 (3)
Workforce	2	1.50 (2)	1.50 (2)	3.00 (2)	5.00 (2)	7.00 (2)	4.00 (2)	6.00 (2)
Materials	2	2.50 (2)	1.50 (2)	2.50 (2)	3.50 (2)	5.50 (2)	6.50 (2)	6.00 (2)
Media	1	2.00 (1)	1.00 (1)	3.00 (1)	5.00 (1)	4.00 (1)	7.00 (1)	6.00 (1)
Sports	1	2.00 (1)	3.00 (1)	1.00 (1)	5.00 (1)	6.00 (1)	4.00 (1)	7.00 (1)
Safety	1	1.00 (1)	4.00 (1)	7.00 (1)	3.00 (1)	6.00 (1)	2.00 (1)	5.00 (1)
(g) Controlled stress								
Num. noise	54	2.19 (53)	2.57 (52)	1.92 (46)	1.63 (54)	1.55 (54)	1.84 (54)	1.83 (54)
Missingness	58	1.90 (57)	1.89 (56)	1.57 (49)	1.66 (58)	1.34 (58)	1.69 (58)	1.71 (58)
Feature mask	58	1.69 (57)	1.58 (56)	1.47 (49)	1.39 (58)	1.35 (58)	1.48 (58)	1.61 (58)
Category shift	30	1.05 (30)	1.03 (30)	1.03 (27)	1.02 (30)	1.02 (30)	1.03 (30)	1.04 (30)
Label noise	58	1.10 (57)	1.10 (56)	1.04 (49)	1.06 (58)	1.07 (58)	1.02 (58)	1.06 (58)
Minority retain	26	1.95 (26)	1.99 (25)	1.80 (21)	2.11 (26)	2.11 (26)	2.84 (26)	2.33 (26)
Num. scale	54	1.58 (53)	1.79 (52)	1.50 (46)	1.41 (54)	1.36 (54)	1.34 (54)	1.43 (54)