TABACE: ROBUST INFERENCE IN TABULAR FOUNDATION MODELS VIA ADAPTIVE CONTEXT ENSEMBLING

Anonymous authors
Paper under double-blind review

ABSTRACT

Tabular Foundation Models (TFMs) have demonstrated remarkable performance across prediction tasks through their in-context learning capabilities. However, perturbations or shifts in the context can easily cause predictions for a query to fluctuate drastically. Leveraging the configurability of TFM contexts, we propose **TabACE** (Adaptive Context Ensembling), a model-agnostic and training-free inference strategy to enhance prediction robustness. TabACE aggregates diverse alternative contexts and modifies the default prediction (Base) only when their agreement and a task-specific gate support the update. We provide a theoretical risk bound guaranteeing that this aggregation and integration effectively limits harmful updates. Across OpenML benchmarks spanning five context perturbations and seven distribution shifts, fully synthetic settings, and natural shifts, TabACE consistently mitigates performance degradation across foundation and conventional predictors while preserving clean performance. Additionally, caching reusable context computations provides a nearly sixfold speedup for repeated inference.

1 INTRODUCTION

Tabular foundation models (TFMs) learn predictive priors through pretraining across tasks and perform predictions via in-context learning without dataset-specific parameter updates (Hollmann et al., 2023; 2025; Qu et al., 2025; Ma et al., 2025). Each prediction depends jointly on the query and its labeled context. Changes in context examples, features, or labels can therefore alter predictions even when the model and query remain unchanged (Abdullah, 2026). This dependence allows TFMs to adapt to new tasks, but also exposes their predictions to unreliable context information. A natural strategy is to evaluate each query under multiple contexts, reducing reliance on any single construction. Such contexts can be obtained through compression, retrieval, sample selection, transformations, and distribution alignment (Feuer et al., 2024; Thomas et al., 2024; Xu et al., 2025; Treerath & Pittorino, 2026; Hollmann et al., 2025; Khalifa et al., 2023). These constructions can provide complementary evidence, but their predictions may disagree and their reliability varies across queries and conditions. Moreover, when the prediction from the original context (*Base*) is already accurate, indiscriminate averaging can make it worse. Improving robustness therefore requires a query-level decision: whether the alternatives provide sufficient evidence to depart from Base, and by how much.

We propose TabACE (Adaptive Context Ensembling), a model-agnostic and query-adaptive inference framework that implements this decision for frozen TFMs. TabACE evaluates complementary context constructions with the same predictor, aggregating their predictions via cross-context agreement and confidence. A task-specific gate then controls the departure from Base, anchoring the final output to Base when alternative evidence is weak. Theoretically, we prove a target risk bound guaranteeing that this evidence-weighted integration effectively limits harmful updates. At runtime, TabACE requires neither ground-truth target labels nor backbone updates, and supports reusable context computations to substantially reduce latency. Across four frozen TFMs on OpenML benchmarks spanning five controlled context perturbations and seven distribution shifts, as well as synthetic and natural shifts, TabACE consistently improves prediction robustness while preserving clean performance. Furthermore, the framework seamlessly transfers to conventional tabular predictors.

The main contributions of this paper are summarized as follows:

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 formulate robust inference for frozen TFMs as deciding when, and by how much, alternative context predictions should modify Base.
- We introduce TabACE, combining complementary contexts, evidence-weighted aggregation, and query-dependent integration with Base, and establish sufficient conditions for risk improvement.
- We demonstrate robustness under controlled perturbations and distribution shifts, supported by synthetic and natural-shift datasets, while preserving clean performance.

2 Related Work

Context Sensitivity in Tabular Foundation Models. Tabular foundation models predict unlabeled queries by conditioning on labeled rows. TabPFN established synthetic-prior inference, TabICL extended tabular ICL to larger datasets, and TabDPT combined retrieval with real-data pretraining (Hollmann et al., 2023; Qu et al., 2025; Ma et al., 2025). Later variants expanded pretraining, scalability, and task coverage (Hollmann et al., 2025; Grinsztajn et al., 2025; QU et al., 2026). Context-sampling studies further show that context size, coverage, and sampling affect prediction accuracy and stability (Abdullah, 2026). These works improve model design and scale, but do not combine predictions from differently constructed contexts under a frozen model.

Context Compression, Retrieval, and Selection. TuneTables, LoCalPFN, MixturePFN, and TabAICL construct inference contexts through compression, retrieval, routing, or active acquisition, respectively (Feuer et al., 2024; Thomas et al., 2024; Xu et al., 2025; Treerath & Pittorino, 2026). TabACE extends context selection to context ensembling: these methods optimize or select a dominant context or expert to improve scalability or query relevance, whereas TabACE integrates predictions induced by heterogeneous context constructors.

Context Ensembling and Construction Diversity. Beyond tabular learning, prior context ensembles combine predictions across partitioned demonstration sets; DENSE, for example, averages their resulting predictions (Khalifa et al., 2023). For tabular prediction, TabPFN ensembles preprocessing and feature or label permutation variants (Hollmann et al., 2025). TabACE extends context ensembling from variants of a shared construction scheme to heterogeneous constructors with distinct inductive biases, assigning their influence from query-level evidence.

Context Variability across Nominal and Shifted Conditions. Robust tabular prediction has been studied under several forms of data change. Classical label-shift methods estimate or recalibrate target class priors (Lipton et al., 2018; Alexandari et al., 2020), while DistPFN, Drift-Resilient TabPFN, and SPN address label, temporal, and strategic shifts in tabular ICL, respectively (Lee, 2026; Helli et al., 2024; Lv et al., 2026). SkillTFM adapts frozen TFMs through evidence-conditioned retrieval of external repair skills; a runtime certificate either admits a repair or falls back to the base predictor (He et al., 2026). TableShift and TabFSBench provide benchmarks for shifted tabular tasks and controlled feature changes (Gardner et al., 2023; Cheng et al., 2025). Context-sampling results additionally show that context choice affects TFM predictions under nominal evaluation (Abdullah, 2026). We therefore study context aggregation under both nominal variability and multiple controlled change families.

3 Method

3.1 Problem Setting and Overview

Let $\mathcal{D}_s = \{(x_j^s, y_j^s)\}_{j=1}^{n_s}$ denote labeled reference data and $\mathcal{X}_t = \{x_i^t\}_{i=1}^{n_t}$ available unlabeled target queries, with $x_j^s, x_i^t \in \mathbb{R}^d$. We consider classification with $\mathcal{Y} = \{1, \dots, L\}$ and regression with $\mathcal{Y} \subseteq \mathbb{R}$. A frozen tabular foundation model f_θ predicts each query from a labeled context constructed from source examples and, for registered selection-shift constructors, pseudo-labeled target examples. Ground-truth target labels are used only for evaluation, and f_θ remains frozen at test time. Reference and target examples follow $P_s(X, Y)$ and $P_t(X, Y)$, respectively. We study both controlled changes to the accessible context and distribution shifts with $P_t \neq P_s$, using $\mathcal{D}_s$, the current query, and any target features required by the registered constructor at inference.

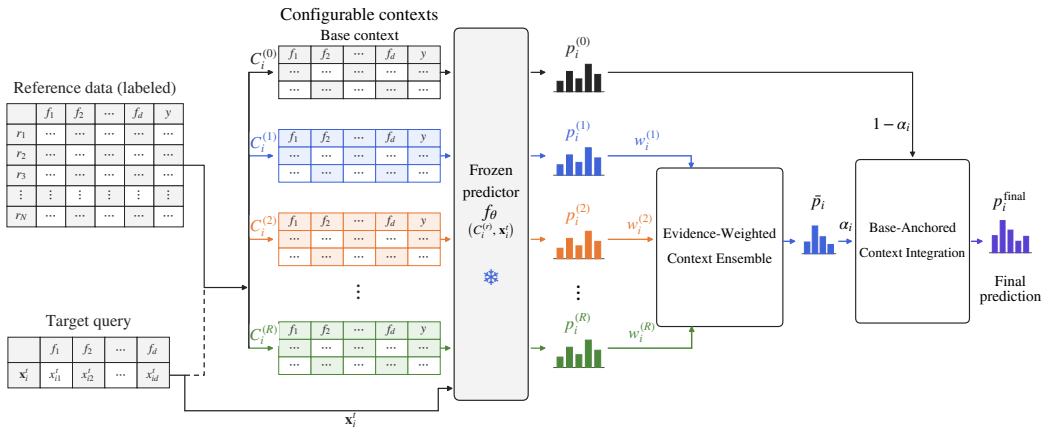


Figure 1: **Framework overview of TabACE.** For each query, configurable constructors form Base and alternative contexts evaluated by a frozen predictor. Evidence weighting aggregates the alternatives, and a Base-anchored gate blends the aggregate with Base into the final prediction. The shown $p_i^{(r)}$ are classification probability vectors; regression uses scalar $\hat{y}_i^{(r)}$ predictions in the same pipeline.

As illustrated in Figure 1, TabACE uses one Base branch and multiple alternative context branches while keeping f_θ frozen. Confidence and cross-context agreement determine the relative influence of the alternatives, while the task-specific gate and aggregate context support set the update magnitude relative to Base. The update therefore follows the available query-level evidence.

3.2 Configurable Context and Output Interface

Let ϕ_0 denote the Base configuration and $\Phi = \{\phi_1, \ldots, \phi_R\}$ the alternative branch configurations chosen from source-side information and application knowledge before target inference. We write $\mathcal{I}(\Phi) = \{1, \ldots, R\}$ for their indices. Let F_{ϕ_r} denote the complete prediction mapping induced by configuration ϕ_r. For an available query x_i^t,

$$o_i^{(r)} = F_{\phi_r}(\mathcal{D}_s, \mathcal{X}_t, x_i^t; f_\theta), \qquad r = 0, \ldots, R. \tag{1}$$

The $\mathcal{X}_t$ argument supplies any target subset required by ϕ_r and is ignored by source-only constructors. For a TFM branch, this mapping composes a configurable context constructor with the frozen predictor. A shared-context constructor forms $C^{(r)} = G_{\phi_r}(\mathcal{D}_s, \mathcal{X}_t)$ once and reuses it across queries, so that $F_{\phi_r}(\mathcal{D}_s, \mathcal{X}_t, x_i^t; f_\theta) = f_\theta(C^{(r)}, x_i^t)$. A query-dependent construction forms $C_i^{(r)} = G_{\phi_r}(\mathcal{D}_s, \mathcal{X}_t, x_i^t)$. A compatible conventional predictor uses the constructed context as its branch-specific training data and exposes the same output interface. For classification, $o_i^{(r)} = p_i^{(r)} \in \Delta^{L-1}$ is a class-probability vector; for point regression, $o_i^{(r)} = \hat{y}_i^{(r)}$ is scalar.

We write $o_i^{(0)}$ for the Base output. Its constructor G_{ϕ_0} follows the backbone's standard sampling rule to form the canonical context from the available labeled reference data, with its selected features and labels unchanged. For $r \geq 1$, G_{ϕ_r} constructs a complementary context by selecting or reweighting reference examples, selecting or transforming features, or applying a prescribed source-derived label adjustment. A selection-shift constructor may also attach Base pseudo-labels to target peers. For query x_i^t, these rows satisfy $\widehat{\mathcal{D}}_{t,-i}^{(0)} \subseteq \{(x_j^t, \widetilde{y}_j^{(0)}) : j \neq i\}$, where $\widetilde{y}_j^{(0)} = \arg\max_{\ell \in \mathcal{Y}} p_{j,\ell}^{(0)}$. The current query is excluded. Constructors use reference information and, when registered, statistics of the available target subset. Target labels remain unavailable to all constructors. When a constructor selects a feature subset, the same projection is applied to the context and the query. The resulting configurations produce the alternative output bank

$$\mathcal{O}_i(\Phi) = \{o_i^{(r)} : r \in \mathcal{I}(\Phi)\}. \tag{2}$$

Here, Φ specifies the context-construction rules and $\mathcal{O}_i(\Phi)$ collects their query-dependent outputs.

3.3 Context Aggregation by Confidence and Agreement

For query x_i^t, the R alternative contexts yield Eq. (2). We derive task-compatible support scores and aggregate the predictions with a shared normalized weighting rule.

For classification, let $p_i^{(r)} \in \Delta^{L-1}$ be the prediction from context r. With source-learned $\eta_{Q,r}, \eta_{S,r} \geq 0$, we compute

$$\begin{aligned} \mu_i &= \frac{1}{R}\sum_{s=1}^{R} p_i^{(s)}, \qquad & Q_i^{(r)} &= 1 - \frac{H(p_i^{(r)})}{\log L}, \\ S_i^{(r)} &= 1 - \frac{1}{2}\|p_i^{(r)} - \mu_i\|_1, \quad & \rho_i^{(r)} &= \eta_{Q,r} Q_i^{(r)} + \eta_{S,r} S_i^{(r)}. \end{aligned} \tag{3}$$

Here $H(p) = -\sum_{\ell=1}^{L} p_\ell \log p_\ell$. $Q_i^{(r)}$ measures normalized-entropy confidence and $S_i^{(r)}$ measures agreement with the mean alternative prediction; $\rho_i^{(r)}$ increases when both are high.

For regression, let $\hat{y}_i^{(r)} \in \mathbb{R}$ be the prediction from context r. We define

$$\begin{aligned} m_i &= \mathrm{median}\{\hat{y}_i^{(s)}\}_{s=1}^{R}, \qquad & S_i^{(r)} &= \exp\left(-\frac{|\hat{y}_i^{(r)} - m_i|}{\tau_y}\right), \\ \rho_i^{(r)} &= \eta_{S,r}^{\mathrm{reg}} S_i^{(r)}, \qquad & \eta_{S,r}^{\mathrm{reg}} &\geq 0. \end{aligned} \tag{4}$$

The median m_i is a robust center and $\tau_y > 0$ is determined from source data; predictions far from this center receive less support.

Both tasks then use the same normalized weighting and aggregation rule:

$$\begin{aligned} w_i^{(r)} &= \frac{\pi_r e^{\rho_i^{(r)}}}{\sum_{s=1}^{R} \pi_s e^{\rho_i^{(s)}}}, \\ \bar{o}_i &= \sum_{r=1}^{R} w_i^{(r)} o_i^{(r)}, \qquad \sum_{r=1}^{R} w_i^{(r)} = 1. \end{aligned} \tag{5}$$

Here $\pi_r > 0$ is the source-learned prior reliability of context r, with $\sum_r \pi_r = 1$. Thus $(o_i^{(r)}, \bar{o}_i) = (p_i^{(r)}, \bar{p}_i)$ for classification and $(o_i^{(r)}, \bar{o}_i) = (\hat{y}_i^{(r)}, \bar{y}_i)$ for regression. The aggregate $\bar{o}_i$ is then integrated with Base.

3.4 Query-Dependent Integration with Base

We integrate $\bar{o}_i$ with Base using a task-specific gate U_i and aggregate context support E_i. Here U_i is intrinsic Base uncertainty for classification and the Base-aggregate discrepancy for regression, while E_i measures alternative support:

$$\begin{aligned} U_i^{\mathrm{cls}} &= \frac{H(p_i^{(0)})}{\log L}, \qquad & E_i^{\mathrm{cls}} &= \sum_{r=1}^{R} w_i^{(r)} Q_i^{(r)} S_i^{(r)}, \\ U_i^{\mathrm{reg}} &= 1 - \exp\left(-\frac{|\bar{y}_i - \hat{y}_i^{(0)}|}{\tau_y}\right), \quad & E_i^{\mathrm{reg}} &= \sum_{r=1}^{R} w_i^{(r)} S_i^{(r)}. \end{aligned} \tag{6}$$

Below, U_i and E_i denote the pair for the current task.

Both tasks use the same integration rule:

$$\alpha_i = \alpha_{\max} U_i E_i, \qquad o_i^{\mathrm{final}} = (1 - \alpha_i) o_i^{(0)} + \alpha_i \bar{o}_i. \tag{7}$$

Since $U_i, E_i, \alpha_{\max} \in [0, 1]$, the update remains close to Base whenever the task-specific gate or context support is weak.

Outer-fold policy learning. Calibration episodes from outer-training datasets fit the context priors, score coefficients, output scale, and integration cap. Each episode pairs a labeled source context with a pseudo-target feature set whose labels enter only the calibration loss. The fitted policy and Φ are fixed before held-out inference.

3.5 THEORETICAL ANALYSIS

The analysis establishes sufficient conditions under which context aggregation improves Base and characterizes how task-gated fusion controls prediction changes according to context support. Let g_0 and g_{final} return $o_i^{(0)}$ and o_i^{final}, respectively, and define $\mathcal{R}_t(g) = \mathbb{E}_{(x,y)\sim P_t}[\ell(g(x), y)]$. For a loss ℓ that is convex in its prediction argument, define the improvements of context r and the context aggregate over Base as

$$\Delta_i^{(r)} = \ell(o_i^{(0)}, y_i^t) - \ell(o_i^{(r)}, y_i^t), \qquad \Delta_i = \ell(o_i^{(0)}, y_i^t) - \ell(\bar{o}_i, y_i^t). \tag{8}$$

These quantities use target labels only for analysis, never for context construction, support scoring, or prediction.

Complementary context coverage. For a change family h, alternative contexts improve Base on different subsets of queries. We use $C_h(\Phi)$ to denote the target probability that at least one context in bank Φ improves Base by at least δ_+. A new context is complementary when it adds beneficial queries not already covered by the bank. Formal definitions and the diminishing-returns property are given in Appendix A.1.

Theorem 1 (Risk bound for weighted context aggregation) *For query i and fixed bank Φ, suppose that a nonempty effective set $\mathcal{G}_i$ satisfies, for some $\delta_+ > 0$, $\delta_- \geq 0$, and $\rho_+ > \rho_-$,*

$$\begin{aligned} &\Delta_i^{(r)} \geq \delta_+, \quad \rho_i^{(r)}(\Phi) \geq \rho_+ \quad (r \in \mathcal{G}_i), \\ &\Delta_i^{(r)} \geq -\delta_-, \quad \rho_i^{(r)}(\Phi) \leq \rho_- \quad (r \in \mathcal{I}(\Phi) \setminus \mathcal{G}_i). \end{aligned} \tag{9}$$

Let $\Pi_i = \sum_{r\in\mathcal{G}_i} \pi_r$ and $W_i = \sum_{r\in\mathcal{G}_i} w_i^{(r)}$. Then

$$\begin{aligned} W_i \geq \underline{W}_i &= \left[1 + \frac{1-\Pi_i}{\Pi_i} e^{-(\rho_+ - \rho_-)}\right]^{-1}, \\ \Delta_i \geq \gamma_i &:= \underline{W}_i \delta_+ - (1 - \underline{W}_i)\delta_-. \end{aligned} \tag{10}$$

Hence $\underline{W}_i > \delta_-/(\delta_+ + \delta_-)$ implies $\gamma_i > 0$ and

$$\ell(o_i^{\text{final}}, y_i^t) - \ell(o_i^{(0)}, y_i^t) \leq -\alpha_i \Delta_i \leq -\alpha_{\max} U_i E_i \gamma_i. \tag{11}$$

Let $\mathcal{A}_\Phi$ contain the queries for which the preceding separation conditions hold and $\gamma_i > 0$. The target-risk difference obeys

$$\mathcal{R}_t(g_{\text{final}}) - \mathcal{R}_t(g_0) \leq -\Gamma_t(\Phi) + \varepsilon_t(\Phi), \tag{12}$$

where

$$\begin{aligned} \Gamma_t(\Phi) &:= \alpha_{\max} \mathbb{E}_{P_t}[U_i E_i \gamma_i \mathbf{1}\{i \in \mathcal{A}_\Phi\}], \\ \varepsilon_t(\Phi) &:= \alpha_{\max} \mathbb{E}_{P_t}[U_i E_i (-\Delta_i)_+ \mathbf{1}\{i \notin \mathcal{A}_\Phi\}]. \end{aligned} \tag{13}$$

Therefore $\Gamma_t(\Phi) > \varepsilon_t(\Phi)$ implies $\mathcal{R}_t(g_{\text{final}}) < \mathcal{R}_t(g_0)$.

Proof sketch. Score separation yields $W_i \geq \underline{W}_i$ for the normalized weight on $\mathcal{G}_i$. Convexity then converts this bound into the aggregate improvement γ_i, which yields the query-level gain in Eq. (11). Splitting the target expectation over $\mathcal{A}_\Phi$ and its complement gives Eq. (12). The complete proof appears in Appendix A.1.

Implications. A larger effective prior mass Π_i or a wider score gap $\rho_+ - \rho_-$ raises $\underline{W}_i$ and hence the guaranteed aggregate gain γ_i. Complementary contexts determine potential beneficial coverage, while score weighting and gating determine how that coverage affects realized risk. The following exact identity gives the magnitude of each task-specific update relative to Base:

$$\|o_i^{\text{final}} - o_i^{(0)}\| = \alpha_{\max} U_i E_i \|\bar{o}_i - o_i^{(0)}\|. \tag{14}$$

Thus, the prediction recovers Base whenever $U_i E_i = 0$. Task-specific classification and regression bounds are given in Appendix A.

Table 1: Overall performance on OpenML-CC18 classification and OpenML-CTR23 regression tasks. The perturbation and shift blocks macro-average all nonzero strengths over five and seven families, respectively.

	Clean				Five Perturbations (Avg.)				Seven Shifts (Avg.)			
	Classification		Regression		Classification		Regression		Classification		Regression	
Method	Acc↑	AUC↑	R^2 ↑	nRMSE↓	Acc↑	AUC↑	R^2 ↑	nRMSE↓	Acc↑	AUC↑	R^2 ↑	nRMSE↓
Non-TFM baselines												
LogReg/Ridge + HPO	0.754	0.867	0.543	0.684	0.418	0.582	-0.559	1.175	0.719	0.775	-0.490	0.851
SVM/SVR + HPO	0.777	0.879	0.492	0.703	0.449	0.611	-0.872	1.255	0.757	0.793	0.010	0.873
kNN + HPO	0.726	0.856	0.587	0.656	0.489	0.647	-1.245	1.396	0.743	0.775	0.327	0.779
RF + HPO	0.776	0.886	0.676	0.561	0.451	0.649	-1.218	1.335	0.760	0.810	0.385	0.724
LightGBM + HPO	0.776	0.884	0.697	0.529	0.463	0.639	-1.215	1.337	0.759	0.796	0.350	0.736
CatBoost + HPO	0.782	0.890	0.708	0.514	0.479	0.653	-1.372	1.368	0.770	0.811	0.381	0.711
XGBoost + HPO	0.774	0.885	0.702	0.525	0.455	0.640	-1.323	1.359	0.751	0.796	0.278	0.767
MLP + HPO	0.729	0.858	0.583	0.568	0.435	0.604	-1.007	1.272	0.741	0.767	0.066	0.828
FT-Transformer	0.745	0.861	0.657	0.568	0.456	0.619	-1.256	1.353	0.716	0.765	0.210	0.822
TabM	0.735	0.866	0.655	0.577	0.454	0.614	-1.100	1.324	0.699	0.743	0.212	0.811
TabulaRNN	0.722	0.856	0.580	0.648	0.454	0.625	-0.851	1.270	0.718	0.754	0.205	0.845
MambaTab	0.748	0.869	0.663	0.564	0.434	0.602	-0.824	1.230	0.751	0.784	0.279	0.761
RealMLP	0.762	0.872	0.694	0.527	0.448	0.614	-1.258	1.326	0.758	0.793	0.240	0.749
LimiX-2M	0.796	0.896	0.724	0.494	0.475	0.657	-1.141	1.277	0.781	0.831	0.385	0.689
+ TabACE-shared	**0.796**	**0.896**	**0.725**	**0.493**	**0.542**	**0.716**	**-0.173**	**1.059**	**0.784**	**0.835**	**0.498**	**0.659**
+ TabACE-specific	**0.796**	**0.896**	**0.725**	**0.493**	**0.628**	**0.769**	**0.053**	**0.947**	**0.804**	**0.847**	**0.629**	**0.563**
TabICL-v2	0.804	0.901	0.739	0.476	0.494	0.676	-1.077	1.270	0.763	0.826	0.273	0.730
+ TabACE-shared	**0.804**	**0.901**	**0.738**	**0.478**	**0.575**	**0.720**	**0.363**	**0.783**	**0.772**	**0.831**	**0.467**	**0.659**
+ TabACE-specific	**0.804**	**0.900**	**0.738**	**0.476**	**0.610**	**0.752**	**0.485**	**0.708**	**0.805**	**0.854**	**0.655**	**0.535**
TabPFN-v3	0.793	0.893	0.739	0.476	0.457	0.636	-1.253	1.313	0.758	0.809	0.426	0.669
+ TabACE-shared	**0.792**	**0.893**	**0.739**	**0.480**	**0.553**	**0.697**	**0.237**	**0.834**	**0.770**	**0.817**	**0.460**	**0.656**
+ TabACE-specific	**0.792**	**0.893**	**0.739**	**0.477**	**0.593**	**0.739**	**0.588**	**0.635**	**0.788**	**0.835**	**0.620**	**0.565**
TabDPT v1.2	0.779	0.891	0.731	0.488	0.448	0.631	-1.152	1.300	0.774	0.821	0.514	0.633
+ TabACE-shared	**0.779**	**0.891**	**0.730**	**0.491**	**0.548**	**0.694**	**0.329**	**0.803**	**0.781**	**0.831**	**0.547**	**0.614**
+ TabACE-specific	**0.779**	**0.891**	**0.730**	**0.488**	**0.600**	**0.746**	**0.443**	**0.733**	**0.805**	**0.846**	**0.610**	**0.572**

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Evaluation tracks. We evaluate TabACE on public OpenML benchmarks and fully synthetic data. The OpenML track draws classification tasks from OpenML-CC18 (Bischl et al., 2021) and regression tasks from OpenML-CTR23 (Fischer et al., 2023). Controlled changes use five-fold dataset-level out-of-fold (OOF) evaluation: policies are fitted on the other four folds and fixed for held-out datasets. Strength zero leaves the context and input unchanged. Controlled data-generating processes in the synthetic track isolate problem family and strength.

Problems and configurations. Both tracks cover five source-context perturbations (Shortcut, Boundary, AUC/Rank-band, Local, and Mixed) and seven distribution shifts (Covariate, Label, Concept, Selection, Domain, Temporal, and Support) at strengths 0, 0.2, 0.5, 0.8, and 1.0. Base is the frozen backbone without context ensembling. Shared uses a fixed context bank across the predefined changes, whereas Specific selects and fuses candidates using a change family specified before target inference. All secondary experiments use frozen TabICL-v2 unless another predictor is named. Base denotes its unmodified prediction. The appendix lists the operators and context layouts.

Task and metrics. We report Accuracy and AUROC for classification and R^2, nMAE, and nRMSE for regression. Robustness rankings and strength-response plots use AUROC and R^2, respectively.

Baselines. The OpenML study compares tuned conventional, tree-based, neural, and frozen foundation-model predictors, including the LimiX-2M checkpoint (hereafter LimiX) (Wang et al., 2026). The synthetic study compares robust-learning, adversarial-training, and distribution-adaptation baselines. The appendix lists all methods and their task applicability, including the compatibility protocol used to cover the complete OpenML suites with LimiX.

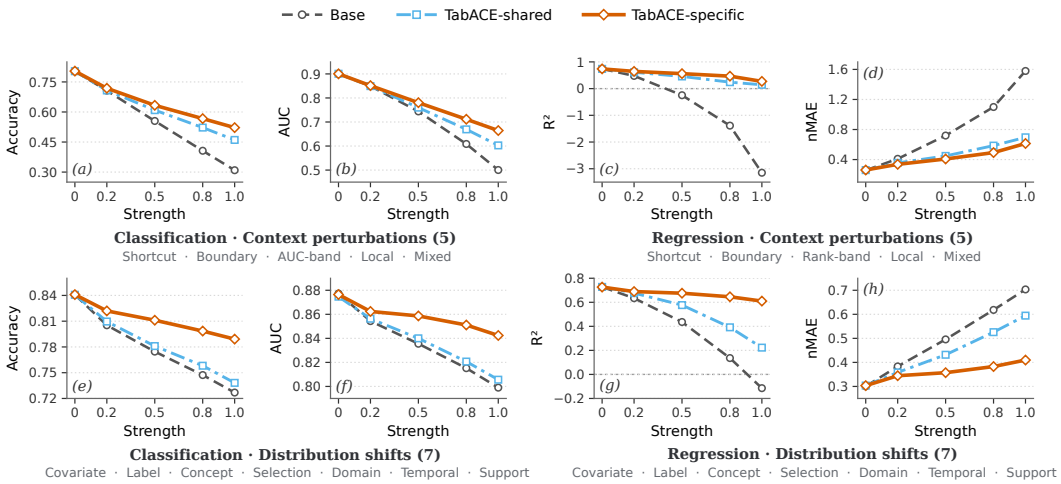


Figure 2: Performance across problem strengths on OpenML benchmarks. Classification panels use OpenML-CC18 and regression panels use OpenML-CTR23. The first row macro-averages five controlled context perturbations; the second macro-averages seven distribution shifts.

Selection protocol. Within each outer fold, context configurations, branch weights, and integration parameters are fitted on calibration episodes from the other four dataset folds and fixed before held-out-dataset evaluation. These episodes cover the registered conditions, but strength is not supplied to the fitted policy or held-out inference. All methods share the same outer folds and evaluation samples, and held-out target labels are used only for final metrics.

4.2 Manipulation Experiments on OpenML Benchmarks

The context-perturbation track contains 72 OpenML-CC18 classification and 35 OpenML-CTR23 regression datasets; the distribution-shift track contains 35 and 33, respectively. Clean performance uses the original context and input.

Table 1 compares thirteen conventional or neural baselines with four frozen tabular foundation models: LimiX-2M, TabICL-v2, TabPFN-v3, and TabDPT v1.2. Relative to Base, Shared improves every nonzero perturbation and shift metric. Specific adds a further gain in every comparison. Clean scores remain essentially unchanged. Baseline configurations, citations, and the full-suite LimiX compatibility protocol appear in Appendix B.3.

Strength response on OpenML tasks. Figure 2 reports all five strengths, equally weighting datasets and the five perturbation or seven shift families. The three configurations are close at strength zero. As strength increases, Base degrades fastest, Shared consistently attenuates the decline, and Specific improves further, achieving the best score across all eight panels at every nonzero strength.

4.3 Fully Synthetic Robustness

Protocol and baselines. We evaluate a fixed five-slot Shared interface on twelve synthetic problem families generated by five data-generating processes with correlated features and controlled targets. Each family is tested at five strengths and three random seeds. Comparisons cover nine robust-learning, adversarial-training, and distribution-adaptation baselines listed in Appendix B.3. Aligned evaluation units use AUROC for classification and R^2 for regression.

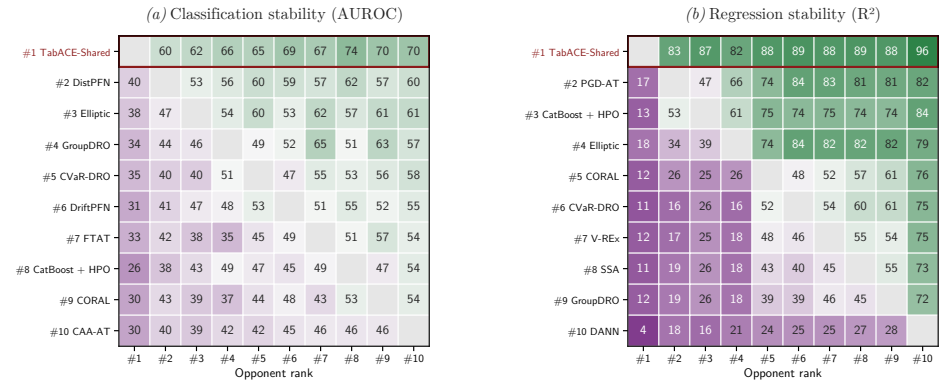


Figure 3: Pairwise stability over 180 aligned seed–DGP–scenario units. Classification uses chance-adjusted AUROC-margin degradation and regression uses normalized R^2 degradation. Cells show how often the row method has the smaller drop; differences within 0.001 are ties.

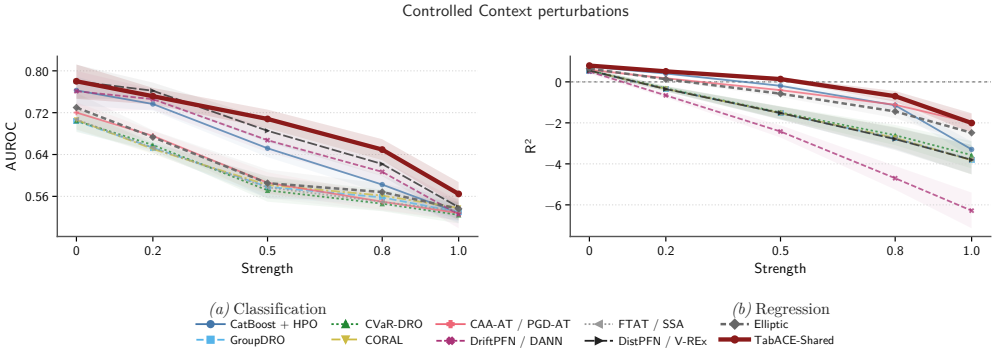


Figure 4: Fully synthetic response to controlled context perturbations. Panels report classification AUROC and regression R^2; curves pool aligned units over five perturbation families and bands show 95% bootstrap intervals.

Stability. For each seed–DGP–scenario unit, we average the four nonzero strengths and normalize classification degradation by the clean AUROC discrimination margin and regression degradation by the clean R^2. Exact definitions appear in Appendix B.1.2. Figure 3 reports pairwise win rates over 180 aligned units. TabACE-Shared ranks first in both classification and regression, with mean win rates of 67.0% and 87.6%, respectively.

Strength response. Figure 4 reports classification AUROC and regression R^2 under controlled context perturbations. Curves equally weight aligned DGP, problem, and seed units and show 95% bootstrap intervals. TabACE-Shared attains the highest classification AUROC from strength 0.5 onward. The context-count and aggregation ablations appear in the appendix.

4.4 Additional Experiments

Natural distribution shifts. Folktables (Ding et al., 2021) provides a 2018 California-to-Michigan geographic shift and a 2014-to-2018 temporal shift within California. Table 2 reports ID and OOD Accuracy for Employment, Income, and TravelTime. Specific achieves the highest score in every setting.

Table 2: ID and OOD Accuracy under natural geographic and temporal shifts.

Method	Domain	Employment	Income	TravelTime
Geographic: California → Michigan (2018)				
Base	ID	0.7922	0.7888	0.5912
	OOD	0.8043	0.7655	0.5472
Shared	ID	0.7948	0.7951	0.5945
	OOD	0.8078	0.7832	0.5961
Specific	ID	**0.7979**	**0.7955**	**0.6002**
	OOD	**0.8089**	**0.7840**	**0.6034**
Temporal: California 2014 → 2018				
Base	ID	0.7704	0.7926	0.5951
	OOD	0.7847	0.7844	0.5908
Shared	ID	0.7751	0.7968	0.6032
	OOD	0.7900	0.7937	0.5975
Specific	ID	**0.7757**	**0.7980**	**0.6061**
	OOD	**0.7916**	**0.7947**	**0.5981**

Table 3: Gains over each unmodified conventional predictor from context ensembling.

Classification: ΔAccuracy ↑

Predictor	Shared	Specific
kNN	+0.0167	**+0.0227**
Logistic	+0.0069	**+0.0287**
CatBoost	+0.0145	**+0.0287**
MLP	+0.0107	**+0.0285**

Regression: ΔR^2 ↑

Predictor	Shared	Specific
kNN	+0.0203	**+0.0654**
Ridge	+0.1425	**+0.1966**
CatBoost	+0.1279	**+0.1740**
MLP	+0.0499	**+0.1645**

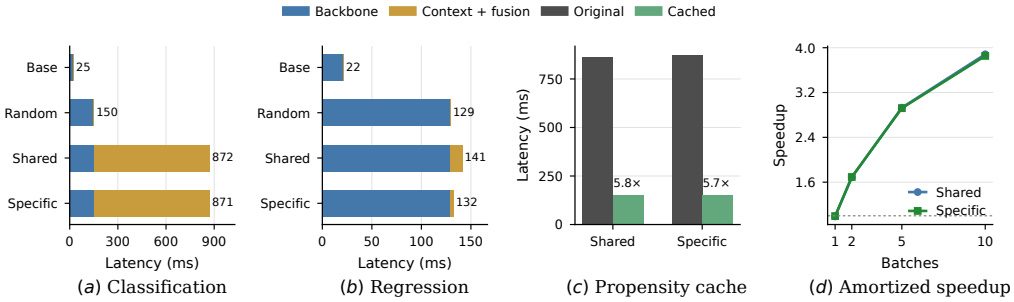


Figure 5: Inference-cost ablation under a fixed budget of five alternative contexts for both Shared and Specific, with a separately evaluated Base branch. Panels (a–b) decompose median latency; (c) compares original and cached inference; (d) includes the one-time cache setup in the amortized speedup.

Conventional-predictor compatibility. Applying the same context-ensemble interface to kNN, Logistic/Ridge, CatBoost, and MLP improves classification Accuracy and regression R^2 for every predictor with Shared, while Specific adds a further gain in all eight comparisons (Table 3).

Inference cost. Figure 5 fixes both Shared and Specific to five alternative contexts for a matched-cost comparison; this coincides with the fixed Shared bank. The figure decomposes wall-clock latency into six backbone passes (Base plus five alternatives) and context construction/fusion. Caching reusable propensity quantities yields about $5.7\times$ faster repeated inference. After including the one-time cache construction, it retains about $3.9\times$ amortized speedup over ten batches.

5 Conclusion

TabACE uses alternative contexts as query-level evidence for a fixed predictor. Consensus weighting identifies mutually supported predictions, while the gate controls how far the output moves from Base. Across controlled benchmarks and natural-shift evaluations, this design improves prediction stability for both foundation and conventional tabular predictors without backbone retraining.

Limitations and future work. The current implementation evaluates multiple candidate contexts for each query. Adaptive pruning could further reduce one-shot latency beyond the caching gains reported here, while evaluation under evolving natural shifts would extend the empirical scope.

Reproducibility Statement

Reproducibility information is organized across the main text and appendix. Appendix A gives the complete theoretical analysis; Appendix B specifies the datasets, splits, perturbations, distribution shifts, context construction, aggregation, and implementation settings; and Appendices C and D report the full strength-wise results and ablations. All experimental results follow the fixed protocol in Appendix B. License-restricted model weights are obtained from their public releases and are not redistributed.

AI Use Statement

The authors used generative AI tools to aid and polish writing, draft portions of the manuscript, and assist with other supporting tasks. All AI-assisted material was reviewed and verified by the authors, who take responsibility for the final manuscript.

References

Mohammed Abdullah. Understanding context sampling in tabpfn on small tabular datasets. *CoRR*, abs/2607.26628, 2026. doi: 10.48550/ARXIV.2607.26628. URL https://doi.org/10.48550/arXiv.2607.26628.

Kazuki Adachi, Shin'ya Yamaguchi, Atsutoshi Kumagai, and Tomoki Hamagami. Test-time adaptation for regression by subspace alignment. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=SXtl7NRyE5.

Md Atik Ahamed and Qiang Cheng. MambaTab: A plug-and-play model for learning tabular data. In *2024 IEEE 7th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 369–375, 2024. doi: 10.1109/MIPR62202.2024.00065.

Amr Alexandari, Anshul Kundaje, and Avanti Shrikumar. Maximum likelihood with bias-corrected calibration is hard-to-beat at label shift adaptation. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 222–232. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/alexandari20a.html.

Bernd Bischl, Giuseppe Casalicchio, Matthias Feurer, Pieter Gijsbers, Frank Hutter, Michel Lang, Rafael Gomes Mantovani, Jan van Rijn, and Joaquin Vanschoren. Openml benchmarking suites. In J. Vanschoren and S. Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/c7e1249ffc03eb9ded908c236bd1996d-Paper-round2.pdf.

Leo Breiman. Random forests. *Machine Learning*, 45:5–32, 2001. doi: 10.1023/A:1010933404324.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi (eds.), *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pp. 785–794. ACM, 2016. doi: 10.1145/2939672.2939785. URL https://doi.org/10.1145/2939672.2939785.

Zi-Jian Cheng, Ziyi Jia, Zhi Zhou, Yu-Feng Li, and Lan-Zhe Guo. TabFSBench: Tabular benchmark for feature shifts in open environments. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 10025–10089. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/cheng25e.html.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995. doi: 10.1007/BF00994018.

Thomas M. Cover and Peter E. Hart. Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory*, 13(1):21–27, 1967. doi: 10.1109/TIT.1967.1053964. URL https://doi.org/10.1109/TIT.1967.1053964.

D. R. Cox. The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2):215–232, 07 1958. ISSN 0035-9246. doi: 10.1111/j.2517-6161.1958.tb00292.x. URL https://doi.org/10.1111/j.2517-6161.1958.tb00292.x.

Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. Retiring adult: New datasets for fair machine learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 6478–6490. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/32e54441e6382a7fbacbbbaf3c450059-Paper.pdf.

Harris Drucker, Christopher J. C. Burges, Linda Kaufman, Alex Smola, and Vladimir Vapnik. Support vector regression machines. In M.C. Mozer, M. Jordan, and T. Petsche (eds.), *Advances in Neural Information Processing Systems*, volume 9. MIT Press, 1996. URL https://proceedings.neurips.cc/paper_files/paper/1996/file/d38901788c533e8286cb6400b40b386d-Paper.pdf.

Anton F Thielmann and Soheila Samiee. On the efficiency of NLP-inspired methods for tabular deep learning. In Mehdi Rezagholizadeh, Peyman Passban, Soheila Samiee, Vahid Partovi Nia, Yu Cheng, Yue Deng, Qun Liu, and Boxing Chen (eds.), *Proceedings of The 4th NeurIPS Efficient Natural Language and Speech Processing Workshop*, volume 262 of *Proceedings of Machine Learning Research*, pp. 532–539. PMLR, 14 Dec 2024. URL https://proceedings.mlr.press/v262/f-thielmann24a.html.

Benjamin Feuer, Robin T Schirrmeister, Valeriia Cherepanova, Chinmay Hegde, Frank Hutter, Micah Goldblum, Niv Cohen, and Colin White. Tunetables: Context optimization for scalable prior-data fitted networks. In *Advances in Neural Information Processing Systems*, volume 37, pp. 83430–83464, 2024.

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.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016. URL http://jmlr.org/papers/v17/15-239.html.

Josh Gardner, Zoran Popovic, and Ludwig Schmidt. Benchmarking distribution shift in tabular data with tableshift. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 53385–53432. Curran Associates, Inc., 2023. doi: 10.52202/075280-2324. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a76a757ed479a1e6a5f8134bea492f83-Paper-Datasets_and_Benchmarks.pdf.

Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 18932–18943. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/9d86d83f925f2149e9edb0ac3b49229c-Paper.pdf.

Yury Gorishniy, Akim Kotelnikov, and Artem Babenko. Tabm: Advancing tabular deep learning with parameter-efficient ensembling. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 77899–77935, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/c1ba41c694834aeef91ae161711d4939-Paper-Conference.pdf.

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

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.

Ali Hasan, Haoming Yang, Yuting Ng, and Vahid Tarokh. Elliptic loss regularization. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=YwzxpZW3p7.

Yi He, Zhengkang Guan, Anpeng Wu, Peng Cui, Fei Wu, and Kun Kuang. Skilltfm: Gated skill evolution for training-free adaptation of tabular foundation models. *CoRR*, abs/2608.06137, 2026. doi: 10.48550/ARXIV.2608.06137. URL https://doi.org/10.48550/arXiv.2608.06137.

Kai Helli, David Schnurr, Noah Hollmann, Samuel Müller, and Frank Hutter. Drift-resilient tabpfn: In-context learning temporal distribution shifts on tabular data. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 98742–98781. Curran Associates, Inc., 2024. doi: 10.52202/079017-3134. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/b2e2774c8e76afe191b5bf518f5cb727-Paper-Conference.pdf.

Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. doi: 10.1080/00401706.1970.10488634. URL https://doi.org/10.1080/00401706.1970.10488634.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/pdf?id=cp5PvcI6w8_.

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.

David Holzmüller, Léo Grinsztajn, and Ingo Steinwart. Better by default: Strong pre-tuned mlps and boosted trees on tabular data. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 26577–26658. Curran Associates, Inc., 2024. doi: 10.52202/079017-0837. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/2ee1c87245956e3eaa71aaba5f5753eb-Paper-Conference.pdf.

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 I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf.

Muhammad Khalifa, Lajanugen Logeswaran, Moontae Lee, Honglak Lee, and Lu Wang. Exploring demonstration ensembling for in-context learning. *CoRR*, abs/2308.08780, 2023. doi: 10.48550/ARXIV.2308.08780. URL https://doi.org/10.48550/arXiv.2308.08780.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (REx). 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. 5815–5826. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/krueger21a.html.

Seunghan Lee. Mitigating label shift in tabular in-context learning via test-time posterior adjustment. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=8N2kUoknvy.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3122–3130. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/lipton18a.html.

Xinpeng Lv, Yunxin Mao, Renzhe Xu, Chunyuan Zheng, Yikai Chen, Haoxuan Li, Jinxuan Yang, Yuanlong Chen, Kun Kuang, Mingyang Geng, Shixuan Liu, Wanrong Huang, Shaowu Yang, Wenjing Yang, Zhouchen Lin, and Haotian Wang. When tabular foundation models meet strategic tabular data: A prior alignment approach. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=jlQeiq1azU.

Junwei Ma, Valentin Thomas, Rasa Hosseinzadeh, Alex Labach, Jesse Cresswell, Keyvan Golestan, Guangwei Yu, Anthony L Caterini, and Maks Volkovs. TabDPT: Scaling tabular foundation models on real data. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 172692–172722. Curran Associates, Inc., 2025. doi: 10.52202/085713-5748. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/fc0e3f908a2116ba529ad0a1530a3675-Paper-Conference.pdf.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=rJzIBfZAb.

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. CatBoost: unbiased boosting with categorical features. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/14491b756b3a51daac41c24863285549-Paper.pdf.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A tabular foundation model for in-context learning on large data. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 50817–50847. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/qu25d.html.

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

R. Tyrrell Rockafellar and Stanislav Uryasev. Optimization of conditional value-at-risk. *The Journal of Risk*, 2(3):21–41, 2000. doi: 10.21314/JOR.2000.038.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986. doi: 10.1038/323533a0.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

Thibault Simonetto, Salah Ghamizi, and Maxime Cordy. Constrained adaptive attack: Effective adversarial attack against deep neural networks for tabular data. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 27817–27849. Curran Associates, Inc., 2024. doi: 10.52202/079017-0873. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/3106a1387b24c33bf66682e7ff393d56-Paper-Conference.pdf.

Baochen Sun and Kate Saenko. Deep CORAL: correlation alignment for deep domain adaptation. In Gang Hua and Hervé Jégou (eds.), *Computer Vision - ECCV 2016 Workshops - Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III*, volume 9915 of *Lecture Notes in Computer Science*, pp. 443–450, 2016. doi: 10.1007/978-3-319-49409-8_35. URL https://doi.org/10.1007/978-3-319-49409-8_35.

Valentin Thomas, Junwei Ma, Rasa Hosseinzadeh, Keyvan Golestan, Guangwei Yu, Maksims Volkovs, and Anthony Caterini. Retrieval & fine-tuning for in-context tabular models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 108439–108467. Curran Associates, Inc., 2024. doi: 10.52202/079017-3442. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/c40daf14d7a6469e65116507c21faeb7-Paper-Conference.pdf.

Wilailuck Treerath and Fabrizio Pittorino. Active in-context learning for tabular foundation models. *CoRR*, abs/2603.27385, 2026. doi: 10.48550/ARXIV.2603.27385. URL https://doi.org/10.48550/arXiv.2603.27385.

Yuanrui Wang, Xingxuan Zhang, Han Yu, Mingchao Hao, Gang Ren, Hao Yuan, Li Mao, Yunjia Zhang, Chun Yuan, and Peng Cui. LimiX-2M: Mitigating low-rank collapse and attention bottlenecks in tabular foundation models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=YjYPYg112A.

Derek Qiang Xu, F. Olcay Cirit, Reza Asadi, Yizhou Sun, and Wei Wang. Mixture of in-context prompters for tabular pfns. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=2fojNANZSv.

Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Seong Joon Oh, Youngjoon Yoo, and Junsuk Choe. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 6022–6031. IEEE, 2019. doi: 10.1109/ICCV.2019.00612. URL https://doi.org/10.1109/ICCV.2019.00612.

Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=r1Ddp1-Rb.

Zhi Zhou, Kun-Yang Yu, Lan-Zhe Guo, and Yu-Feng Li. Fully test-time adaptation for tabular data. In Toby Walsh, Julie Shah, and Zico Kolter (eds.), *Thirty-Ninth AAAI Conference on Artificial Intelligence, Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence, Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, pp. 23027–23035. AAAI Press, 2025. doi: 10.1609/AAAI.V39I21.34466. URL https://doi.org/10.1609/aaai.v39i21.34466.

Appendix Contents

A Theoretical Analysis 16

A.1 Proof of Theorem 1 16

A.2 Theoretical Robustness 17

B Experimental Setup and Reproducibility 19

B.1 Evaluation Protocol 19

B.2 Controlled Perturbations and Distribution Shifts 20

B.3 Compared Methods and Task Applicability 22

B.4 Context Construction 24

B.5 Implementation Details 29

C Backbone-Wise Strength Results 29

C.1 TabICL-v2 30

C.2 Additional Backbones 31

D Ablation Studies 33

D.1 Context Budget 33

D.2 Aggregation Rule 33

D.3 Fixed-Bank Component Ablation 34

D.4 Gate Behavior across Evidence Bins 34

A Theoretical Analysis

A.1 Proof of Theorem 1

Analysis setup and notation. Let ϕ_0 be the Base configuration and $\Phi = \{\phi_1, \ldots, \phi_R\}$ the alternative configuration set from Section 3.2, indexed by $\mathcal{I}(\Phi) = \{1, \ldots, R\}$. For query i, context r produces $o_i^{(r)}$, receives the source-calibrated prior π_r, and has bank-dependent label-free score $\rho_i^{(r)}(\Phi)$, abbreviated below as $\rho_i^{(r)}$. Equation (5) forms weights $w_i^{(r)}$ and the aggregate $\bar{o}_i$. The final prediction is the task-gated mixture

$$o_i^{\text{final}} = (1 - \alpha_i) o_i^{(0)} + \alpha_i \bar{o}_i, \qquad \alpha_i = \alpha_{\max} U_i E_i,$$

where $U_i, E_i, \alpha_{\max} \in [0, 1]$.

For a loss convex in its prediction argument, the context-wise and aggregate improvements are

$$\Delta_i^{(r)} = \ell(o_i^{(0)}, y_i^t) - \ell(o_i^{(r)}, y_i^t), \qquad \Delta_i = \ell(o_i^{(0)}, y_i^t) - \ell(\bar{o}_i, y_i^t).$$

Target labels appear only in these analysis quantities. The construction, weighting, and Base integration remain label-free at target inference. We write $P_t^{(h)}$ for the target distribution associated with change family h and P_t for the target distribution used in the risk analysis.

Context complementarity and beneficial coverage. For fixed $\delta_+ > 0$, define

$$\mathcal{H}_r^h = \{i : \Delta_i^{(r)} \geq \delta_+\}, \qquad C_h(\Phi) = \Pr_{P_t^{(h)}}\left(i \in \bigcup_{r \in \mathcal{I}(\Phi)} \mathcal{H}_r^h \right). \tag{15}$$

The set $\mathcal{H}_r^h$ collects the queries that context r can improve under family h and is fixed once its construction and branch prediction are fixed. The union measures which queries have at least one beneficial context.

Adding a configuration $\phi_{r'}$ changes this coverage by

$$C_h(\Phi \cup \{\phi_{r'}\}) - C_h(\Phi) = \Pr_{P_t^{(h)}}\left(i \in \mathcal{H}_{r'}^h \setminus \bigcup_{r \in \mathcal{I}(\Phi)} \mathcal{H}_r^h \right). \tag{16}$$

Lemma A.1 (Monotone and submodular context coverage) *The coverage function C_h is monotone and submodular. For configuration sets $\Phi_1 \subseteq \Phi_2$ and $\phi_{r'} \notin \Phi_2$,*

$$\begin{aligned} 0 &\leq C_h(\Phi_2 \cup \{\phi_{r'}\}) - C_h(\Phi_2) \\ &\leq C_h(\Phi_1 \cup \{\phi_{r'}\}) - C_h(\Phi_1). \end{aligned} \tag{17}$$

Proof. Let $\mathcal{U}_j = \bigcup_{r \in \mathcal{I}(\Phi_j)} \mathcal{H}_r^h$. The marginal coverage of r' is

$$\Pr_{P_t^{(h)}}(i \in \mathcal{H}_{r'}^h \setminus \mathcal{U}_j).$$

Since $\mathcal{U}_1 \subseteq \mathcal{U}_2$, the set difference for Φ_2 is contained in that for Φ_1. Both inequalities follow. □

Equation (17) characterizes potential beneficial coverage, not realized aggregate risk. Bank-dependent scores and gating connect coverage to risk through $\mathcal{G}_i$ below.

Score separation and effective weight. Fix a bank Φ, query i, and a nonempty effective set $\mathcal{G}_i$. Under Eq. (9), let

$$N_{\mathcal{G}} = \sum_{r \in \mathcal{G}_i} \pi_r e^{\rho_i^{(r)}}, \qquad N_{\mathcal{B}} = \sum_{r \in \mathcal{I}(\Phi) \setminus \mathcal{G}_i} \pi_r e^{\rho_i^{(r)}}.$$

Writing $\Pi_i = \sum_{r \in \mathcal{G}_i} \pi_r$, score separation gives

$$N_{\mathcal{G}} \geq \Pi_i e^{\rho_+}, \qquad N_{\mathcal{B}} \leq (1 - \Pi_i) e^{\rho_-}.$$

The normalized mass on effective contexts consequently satisfies

$$W_i = \frac{N_{\mathcal{G}}}{N_{\mathcal{G}} + N_{\mathcal{B}}} \geq \left[1 + \frac{1 - \Pi_i}{\Pi_i} e^{-(\rho_+ - \rho_-)}\right]^{-1} = \underline{W}_i. \tag{18}$$

This bound exposes two routes to a larger effective weight: greater prior mass Π_i and a wider score gap $\rho_+ - \rho_-$. Convexity then yields

$$\begin{aligned} \Delta_i &\geq \sum_{r=1}^{R} w_i^{(r)} \Delta_i^{(r)} \\ &\geq W_i \delta_+ - (1 - W_i)\delta_- \\ &\geq \underline{W}_i \delta_+ - (1 - \underline{W}_i)\delta_- = \gamma_i. \end{aligned} \tag{19}$$

Hence the effective contexts dominate the aggregate whenever $\underline{W}_i > \delta_-/(\delta_+ + \delta_-)$.

The scores combine source calibration with target-side confidence, agreement, or proximity. Their joint ordering determines whether the separation condition holds; the analysis below also covers imperfect ordering.

Proof of Theorem 1. Equation (18) proves the lower bound on W_i, and Eq. (19) proves $\Delta_i \geq \gamma_i$. The condition $\underline{W}_i > \delta_-/(\delta_+ + \delta_-)$ is equivalent to $\gamma_i > 0$.

Applying convexity to the task-gated update gives

$$\begin{aligned} \ell(o_i^{\text{final}}, y_i^t) &\leq (1 - \alpha_i)\ell(o_i^{(0)}, y_i^t) + \alpha_i \ell(\bar{o}_i, y_i^t), \\ \ell(o_i^{\text{final}}, y_i^t) - \ell(o_i^{(0)}, y_i^t) &\leq -\alpha_i \Delta_i. \end{aligned}$$

For $i \in \mathcal{A}_\Phi$, the aggregate bound therefore gives

$$\ell(o_i^{\text{final}}, y_i^t) - \ell(o_i^{(0)}, y_i^t) \leq -\alpha_{\max} U_i E_i \gamma_i.$$

For $i \notin \mathcal{A}_\Phi$, the same convexity inequality gives

$$\ell(o_i^{\text{final}}, y_i^t) - \ell(o_i^{(0)}, y_i^t) \leq \alpha_{\max} U_i E_i (-\Delta_i)_+.$$

Taking the P_t expectation and splitting it over $\mathcal{A}_\Phi$ and its complement yields Eq. (12). The strict risk improvement follows when $\Gamma_t(\Phi) > \varepsilon_t(\Phi)$. □

Interpretation: family-directed construction. Family-directed construction enters through complementary coverage and effective score mass. A construction adds coverage when its beneficial region contains queries outside the union of the current regions, as quantified by Eq. (16). On a covered query, a larger effective prior mass Π_i or score gap $\rho_+ - \rho_-$ increases the lower bound on W_i when the separation condition holds, increasing the margin γ_i.

These results connect family-directed construction, complementary coverage, score weighting, and the task-specific gate to the risk decomposition in Theorem 1. Deviations from score separation and harmful high-support consensus are analyzed in Appendix A.2.

A.2 Theoretical Robustness

Risk under imperfect score separation. Let $\mathcal{A}_\Phi$ be the beneficial-update region defined in Theorem 1. Its complement contains queries for which the required improvement, score ordering, or aggregate margin does not hold. Define its gate-weighted mass by

$$M_t(\Phi) := \mathbb{E}_{P_t}[U_i E_i \mathbf{1}\{i \notin \mathcal{A}_\Phi\}]. \tag{20}$$

Proposition A.2 (Imperfect score separation) *Suppose $(-\Delta_i)_+ \leq L_\Delta < \infty$ on the target prediction domain. This holds, for example, when each class probability satisfies $p_k \geq \varepsilon_p > 0$ and regression targets and predictions are bounded. Then*

$$\mathcal{R}_t(g_{\text{final}}) - \mathcal{R}_t(g_0) \leq -\Gamma_t(\Phi) + \alpha_{\max} L_\Delta M_t(\Phi). \tag{21}$$

Proof. The residual term in Eq. (13) satisfies

$$\varepsilon_t(\Phi) \le \alpha_{\max} L_\Delta \mathbb{E}_{P_t}[U_i E_i \mathbf{1}\{i \notin \mathcal{A}_\Phi\}].$$

Substitution into Eq. (12) proves the result. □

Proposition A.2 requires separation on the beneficial portion of the target distribution rather than on every query. Score-ordering errors enter through $M_t(\Phi)$ and are attenuated by the same task-specific and context-support gates used by the algorithm.

Harmful high-support consensus. Correlated alternative errors can produce high agreement around an adverse aggregate. For a threshold $e \in [0, 1]$, define the harmful high-support consensus rate

$$\eta_{\text{fc}}(e) := \Pr_{P_t}(E_i > e,\ \Delta_i < 0). \tag{22}$$

This event isolates confident agreement that moves the aggregate away from the target label under the analyzed loss.

Proposition A.3 (Residual-risk refinement) *Under the bounded adverse improvement condition of Proposition A.2,*

$$\varepsilon_t(\Phi) \le \alpha_{\max} L_\Delta \left[e + \eta_{\text{fc}}(e)\right]. \tag{23}$$

Consequently,

$$\mathcal{R}_t(g_{\text{final}}) - \mathcal{R}_t(g_0) \le -\Gamma_t(\Phi) + \alpha_{\max} L_\Delta \left[e + \eta_{\text{fc}}(e)\right]. \tag{24}$$

Proof. Dropping the indicator of $\mathcal{A}_\Phi^c$ only enlarges the residual. Split the remaining expectation at $E_i = e$. On $E_i \le e$, $U_i E_i(-\Delta_i)_+ \le e L_\Delta$. On $E_i > e$, a positive contribution requires $\Delta_i < 0$, while $U_i E_i(-\Delta_i)_+ \le L_\Delta$. Taking expectations gives Eq. (23); combining it with Eq. (12) gives Eq. (24). □

Equation (23) is a conservative decoupled bound obtained with $U_i \le 1$; Eq. (13) retains U_i.

The bound divides residual harm into low-support updates, controlled by e, and harmful high-support consensus, controlled by $\eta_{\text{fc}}(e)$. Complementary contexts can enlarge $\Gamma_t(\Phi)$, whereas repeated adverse agreement contributes to $\eta_{\text{fc}}(e)$.

Base-deviation bounds. The update in Eq. (7) satisfies the exact identity

$$o_i^{\text{final}} - o_i^{(0)} = \alpha_{\max} U_i E_i (\bar{o}_i - o_i^{(0)}).$$

Thus the final prediction recovers Base whenever $U_i E_i = 0$, and otherwise its displacement is scaled by the product of the two gates.

For classification,

$$p_i^{\text{final}} - p_i^{(0)} = \alpha_i (\bar{p}_i - p_i^{(0)}).$$

The homogeneity of total variation, together with $\text{TV}(\bar{p}_i, p_i^{(0)}) \le 1$ and $E_i^{\text{cls}} \le 1$, yields

$$\text{TV}(p_i^{\text{final}}, p_i^{(0)}) \le \alpha_i \le \alpha_{\max} \frac{H(p_i^{(0)})}{\log L}. \tag{25}$$

Let $M_i^{(0)}$ be the gap between the largest and second-largest Base class probabilities. Since each probability changes by at most $\alpha_i \|\bar{p}_i - p_i^{(0)}\|_\infty$, the predicted class remains unchanged whenever

$$2\alpha_i \|\bar{p}_i - p_i^{(0)}\|_\infty < M_i^{(0)}. \tag{26}$$

For regression, U_i is the Base-aggregate discrepancy gate. Let $d_i = |\bar{y}_i - \hat{y}_i^{(0)}|$. Since $E_i^{\text{reg}} \le 1$ and $1 - e^{-x} \le x$ for $x \ge 0$,

$$\begin{aligned} |\hat{y}_i^{\text{final}} - \hat{y}_i^{(0)}| &= \alpha_i d_i \\ &\le \alpha_{\max} \left(1 - e^{-d_i/\tau_y}\right) d_i \\ &\le \frac{\alpha_{\max}}{\tau_y} d_i^2. \end{aligned} \tag{27}$$

Equations (25)–(27) are the task-specific forms of the Base-deviation identity in Eq. (14).

B Experimental Setup and Reproducibility

B.1 Evaluation Protocol

B.1.1 Benchmarks and Evaluation Splits

We use complementary public-data and fully synthetic evaluation tracks. The public-data track further separates five controlled context perturbations from seven distribution shifts. Table 4 summarizes the fixed task sets and sample protocols.

Table 4: Public-data benchmark composition and fixed sample protocols. A cap is an upper bound; smaller official folds retain all available rows.

Track	Classification	Regression
Controlled context perturbations	All 72 OpenML-CC18 tasks, including binary and multiclass tasks. Each task uses 384 fixed rows: 256 labeled source-context rows and 128 target evaluation rows.	All 35 OpenML-CTR23 tasks. The official source and target partitions are capped at 2,048 and 512 rows, respectively.
Distribution shifts	A fixed 35-task binary subset of OpenML-CC18. Each task uses 256 historical source rows, 64 reference rows, and 128 target evaluation rows, with a labeled context budget of 256.	The 33 OpenML-CTR23 tasks that support the fixed partition: at most 1,024 source rows and 256 target rows, with 256 current-context rows, 128 reference rows, and the remaining source rows used as history.

The distribution-shift regression track excludes `forest_fires` and `student_performance_por`. Their official training folds contain fewer than 640 rows and therefore cannot support disjoint current, reference, and historical partitions under the fixed budget.

Public-data policy selection uses five-fold *dataset-level* out-of-fold (OOF) evaluation. The outer folds partition complete datasets rather than rows. For each held-out dataset, context configurations and the fusion policy are determined using datasets in the other four folds and then frozen; only preprocessing and constructor statistics are recomputed from its source context. All methods share the same dataset folds and base source–target partition. For each problem family, we evaluate strengths $s \in \{0, 0.2, 0.5, 0.8, 1.0\}$; $s = 0$ preserves the original context and inputs. Within a sweep, feature maps retain target row identities, while the label-shift operator resamples complete rows from fixed class-specific target pools.

The fully synthetic track uses five response mechanisms—linear, interaction, smooth nonlinear, threshold, and subgroup. Seed 0 is used for method development, whereas the reported confirmation results use previously unseen seeds 4, 5, and 6. Each condition contains 512 source rows and 384 target rows with a fixed 256-row labeled context. Within a seed–DGP–scenario unit, all methods and strengths reuse the same base samples, random streams, and context indices. The five DGPs, three confirmation seeds, and twelve non-clean problem families yield $5 \times 3 \times 12 = 180$ aligned evaluation units. Section B.2 specifies the generators and transformations.

B.1.2 Metrics and Statistical Reporting

Classification is evaluated with Accuracy and AUROC, and regression with R^2, normalized mean absolute error (nMAE), and normalized root mean squared error (nRMSE). Higher is better for Accuracy, AUROC, and R^2; lower is better for nMAE and nRMSE. AUROC and R^2 are the primary metrics for the robustness rankings and strength–response analyses.

For target values y_i and predictions $\hat{y}_i$, let

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|, \qquad \mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}. \tag{28}$$

We report

$$\mathrm{nMAE} = \frac{\mathrm{MAE}}{\max(\sigma, 10^{-12})}, \qquad \mathrm{nRMSE} = \frac{\mathrm{RMSE}}{\max(\sigma, 10^{-12})}, \tag{29}$$

where the frozen benchmark implementations use

$$\sigma = \begin{cases} \mathrm{std}(y_{\mathrm{source}}), & \text{controlled context perturbations,} \\ \mathrm{std}(y_{\mathrm{target}}), & \text{distribution shifts.} \end{cases} \tag{30}$$

Both are population standard deviations (`ddof=0`). Because the two tracks use different normalization scales, nMAE and nRMSE are compared only within a track; cross-track regression summaries use R^2. In the shift track, $\mathrm{std}(y_{\mathrm{target}})$ is computed only after all predictions have been produced and is used solely for metric evaluation.

For public data, we compute one dataset-level OOF score per condition and then macro-average across datasets, giving every dataset equal weight. A summary over several problem families first averages dataset-level scores within each family and strength and then takes an equal-weight mean over the five context perturbations or seven distribution shifts. Reported public-data curves use only observed points, without smoothing, fitting, or monotonic projection.

Synthetic confidence intervals use 10,000 nonparametric bootstrap resamples with replacement and the 2.5th and 97.5th percentiles of the bootstrap means. The resampling unit is a complete seed–DGP–scenario unit rather than an individual row. The clean anchor contains $3 \times 5 = 15$ units; each nonzero controlled-perturbation point contains $3 \times 5 \times 5 = 75$ units; each nonzero distribution-shift point contains $3 \times 5 \times 7 = 105$ units; and the twelve-family summary contains 180 units.

For the pairwise stability analysis, we average each metric over the four nonzero strengths within a seed–DGP–scenario unit. Writing AUROC as AUC, the normalized classification and regression degradations are

$$d_{\mathrm{cls}} = \frac{\mathrm{AUC}_{\mathrm{clean}} - \mathrm{AUC}_{\mathrm{shift}}}{\max(\mathrm{AUC}_{\mathrm{clean}} - 0.5, 0.05)}, \qquad d_{\mathrm{reg}} = \frac{R^2_{\mathrm{clean}} - R^2_{\mathrm{shift}}}{\max(|R^2_{\mathrm{clean}}|, 0.1)}. \tag{31}$$

For each method pair, the smaller degradation counts as a win; absolute differences no greater than 0.001 are counted as ties.

B.1.3 Leakage Control

The dataset-level OOF design separates policy selection from held-out-dataset evaluation. For a held-out dataset, fusion weights and other data-dependent policy parameters are determined from the other four folds and then frozen. Constructors may use the labeled source context available to the method, and target-aware constructors may additionally inspect unlabeled target features X_t, but never target labels.

Held-out labels are not used for context construction, candidate selection, fusion-policy fitting, or prediction. They are accessed only after prediction to compute the reported metrics. Metadata visibility is fixed by the method interface: Shared uses a family-agnostic context registry, whereas the prior-informed Specific variant may receive the problem-family identifier specified before target inference. Section B.4.6 details candidate selection and aggregation for each interface.

B.2 Controlled Perturbations and Distribution Shifts

B.2.1 Common Design

We express every intervention as a deterministic, severity-indexed operator $\mathcal{T}_s$, where $s \in \{0, 0.2, 0.5, 0.8, 1.0\}$ and $\mathcal{T}_0$ is the identity. For a source-context perturbation, s determines the affected-row count $m_s = \lceil s n_c \rceil$ for $s > 0$; for a distribution shift, it controls either a nested replacement fraction or the magnitude of a feature or response map. Within a dataset–family pair, the clean partition and base cohort are fixed, and every random seed is a deterministic function of the registered condition. No split, model, or policy is retuned along a strength curve.

The synthetic track draws $X \sim \mathcal{N}(0, \Sigma)$ in 12 dimensions, where $\Sigma_{jj} = 1$, $\Sigma_{12} = \Sigma_{21} = 0.35$, $\Sigma_{34} = \Sigma_{43} = -0.25$, and all other off-diagonal entries are zero. It uses five fixed response mechanisms—linear, pairwise interaction, smooth nonlinear, threshold, and two-subgroup piecewise linear—with the same active coordinates and coefficients across all methods, strengths, and seeds. For each generated block, $g(x) = h(x) - \mathrm{median}\{h(X)\}$. Classification draws $Y \sim \mathrm{Bernoulli}(\mathrm{sigmoid}(g(X)))$; regression uses $Y = g(X) + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.65^2)$. The

rotated concept rule reverses the first five signal coordinates. The public-data track starts from the fixed OpenML partitions in Table 4 and applies empirical resampling or feature/response transformations. Consequently, distributional equalities below describe the synthetic construction; their OpenML counterparts preserve the corresponding variables or row identities by design, rather than asserting exact finite-sample equality.

B.2.2 Controlled Source-Context Perturbations

Let $j^\star$ denote the feature with the largest absolute source feature–target association. For each family, a deterministic rule selects a subset I_s of m_s source rows; score-ranked families are nested across severity, whereas Mixed uses a seeded subset at each severity. Only (X_c, y_c) on these rows is modified; the target evaluation cohort and its labels are unchanged. Table 5 gives the complete operator-level description.

Table 5: Controlled transformations of the labeled source context. Label cycling applies only to classification; opposite-tail response replacement applies only to regression.

Family	Affected-row rule	Transformation on selected rows
Shortcut	Decreasing absolute label/response deviation from its median.	Replace $X_{j^\star}$ by a strong value on the side opposite to the observed label or response; y is retained.
Boundary	Increasing distance to empirical label/response quartile cut points.	Classification shifts $X_{j^\star}$ against the label and cycles a deterministic half of the labels; regression sends responses to the opposite empirical tail.
AUC/Rank-band	Increasing distance to the median of $X_{j^\star}$.	Classification shifts $X_{j^\star}$ and cycles every third selected label; regression sends selected responses to the opposite empirical tail.
Local	Increasing distance to a fixed local anchor in feature space.	Modify both $X_{j^\star}$ and the label/response, creating a contradictory pattern confined to one neighborhood.
Mixed	A seeded random subset of size m_s.	Partition the selected subset into four fixed chunks and apply Shortcut, Boundary, AUC/Rank-band, and Local, respectively.

B.2.3 Distribution-Shift Families

For the synthetic track, let Z denote latent features, g_{rot} a response rule with reordered signal coordinates, and T_s a severity-controlled observation map. Table 6 first states the intended distributional change and then records its classification and regression OpenML realizations. The public classification protocol can transform the historical/reference source blocks or the target block, whereas the public regression protocol transforms the current labeled context against a fixed target cohort; this distinction follows the sample interfaces in Table 4.

Table 6: Seven source–target shift families. The invariant column describes the synthetic population construction; the last two columns specify the empirical OpenML operators.

Family	Synthetic change / invariant	OpenML classification	OpenML regression
Covariate	Change $P(X)$ while retaining $P(Y \mid X)$.	Resample historical/reference rows with an X-only weight; keep the target cohort fixed.	Replace a nested fraction of current-context rows along the highest-variance feature; keep the target fixed.
Label	Change $P(Y)$ while retaining $P(X \mid Y)$.	Resample complete target rows within class to vary the class prior.	Resample complete current-context rows toward one response tail.
Concept	Interpolate g toward g_{rot} at fixed $P(X)$.	Apply nested local boundary-label changes to reference and target rows; features are fixed.	Add a smooth, feature-dependent response drift to the current context; features are fixed.
Selection	Make source inclusion depend on X; retain $P(Y \mid X)$.	Resample source rows by an X-only predicted-majority affinity; keep the target fixed.	Resample current rows by X-only distance from the target center.
Domain	Use T_0 and T_s on shared latent rows; retain $P(Y \mid Z)$.	Apply an affine observation map to target features; row identities and labels are fixed.	Apply an affine observation map to current-context features; responses are fixed.
Temporal	Evolve both T_s and the response rule while preserving temporal order.	Apply joint feature and local concept drift to the reference and target blocks; historical rows remain earlier observations.	Apply joint feature and response drift to the current block while retaining disjoint history and reference blocks.
Support	Translate observed support while retaining $P(Y \mid Z)$.	Translate target coordinates; row identities and labels are fixed.	Resample the current context toward a restricted central support using an X-only score.

All intervention operators are created before method evaluation. Target labels used to define or score a benchmark condition are never exposed to a compared method, a context constructor, or the aggregation policy.

B.3 Compared Methods and Task Applicability

We organize the compared methods by evaluation purpose. The OpenML study compares standard statistical, tree-based, deep tabular, and frozen foundation-model predictors under common data partitions. The synthetic study focuses on methods designed for perturbations or distribution shifts. Because not every robustness method supports both prediction tasks, we distinguish shared, classification-only, and regression-only baselines. This subsection defines the comparison set and task applicability; protocol and selection details are separated into the surrounding appendix subsections.

B.3.1 OpenML Baselines

Classical and tree models.

- **Logistic/Ridge**—regularized linear classification and regression (Cox, 1958; Hoerl & Kennard, 1970).
- **SVM/SVR**—margin-based classification and regression (Cortes & Vapnik, 1995; Drucker et al., 1996).
- **kNN**—classification or regression from local neighbors in feature space (Cover & Hart, 1967).
- **Random Forest**—bagged decision-tree ensembles (Breiman, 2001).
- **LightGBM**—histogram-based gradient-boosted trees (Ke et al., 2017).
- **CatBoost**—gradient-boosted trees trained with ordered boosting (Prokhorenkova et al., 2018).
- **XGBoost**—regularized gradient-boosted trees (Chen & Guestrin, 2016).
- **MLP**—a feed-forward neural baseline trained on each source fold (Rumelhart et al., 1986).

Deep tabular models.

- **FT-Transformer**—a Transformer over tokenized tabular features (Gorishniy et al., 2021).
- **TabM**—a parameter-efficient ensemble of MLP-like predictors (Gorishniy et al., 2025).
- **TabulaRNN**—an RNN-inspired architecture for efficient tabular learning (F Thielmann & Samiee, 2024).

- **MambaTab**—a structured state-space architecture for tabular prediction (Ahamed & Cheng, 2024).
- **RealMLP**—an MLP with a pre-tuned configuration for tabular data (Holzmüller et al., 2024).

Frozen foundation-model backbones.

- **LimiX-2M**—the official checkpoint, evaluated without parameter updates and displayed under its full model name in the main table (Wang et al., 2026).
- **TabICL-v2**—a scalable tabular in-context learning model (QU et al., 2026).
- **TabPFN-v3**—a pretrained tabular foundation model (Grinsztajn et al., 2026).
- **TabDPT v1.2**—a tabular foundation model pretrained on real data (Ma et al., 2025).

LimiX compatibility on the complete suites. The LimiX-2M study excludes classification datasets with more than ten target classes and consequently reports 62 of the 72 OpenML-CC18 tasks; applying the same filters leaves 33 OpenML-CTR23 tasks (Wang et al., 2026). Our main table instead targets the complete 72-task CC18 and 35-task CTR23 context-perturbation suites. Of the 72 classification tasks, 67 use the native multiclass interface. The remaining five use a predeclared one-vs-rest (OvR) compatibility wrapper: `letter` (26 classes), `vowel` (11), `isolet` (26), `texture` (11), and `Devnagari-Script` (46). Each class is evaluated by one binary LimiX call, and the positive-class probabilities are normalized across classes for Accuracy and macro one-vs-rest AUROC. These five results are wrapper-based rather than native multiclass LimiX results, and the checkpoint remains fixed.

For the three CC18 tasks with more than 1,024 predictor columns: `Bioresponse` (1,776), `CIFAR_10` (3,072), and `Internet-Advertisements` (1,558), the perturbation is first applied in the original feature space. A deterministic TruncatedSVD, fitted only on the unmodified labeled source features, then maps both context and target features to at most 256 components. Target labels are never used by this projection. All 35 regression tasks use the native LimiX regression interface without OvR or feature projection. The official non-retrieval ensembles use four classification and eight regression members; all configurations and pretrained parameters remain fixed across clean and nonzero-strength conditions.

The seven-shift classification block follows the frozen 35-task binary CC18 subset used by that protocol. The network loader converts nominal features to ordinal codes and median-imputes missing feature values before forming the split. We apply the same deterministic conversion to cached OpenML parquet files before invoking the unchanged split and shift constructors; the conversion uses no target labels.

The foundation models retain their prescribed prediction interfaces and their pretrained parameters are not updated. For each frozen backbone, Base, TabACE-Shared, and the prior-informed TabACE-Specific use the same data partitions, context budget, and target evaluation examples.

B.3.2 Synthetic Robustness Baselines

Shared classification and regression baselines.

- **CatBoost**—a boosted-tree reference that does not depend on a task-specific neural architecture (Prokhorenkova et al., 2018).
- **GroupDRO**—optimization against the worst predefined group (Sagawa et al., 2020).
- **CVaR-DRO**—optimization of the upper tail of the loss distribution (Rockafellar & Uryasev, 2000).
- **CORAL**—source–target covariance alignment (Sun & Saenko, 2016).
- **Elliptic Loss Regularization**—loss regularization for robust prediction (Hasan et al., 2025).

Classification-only baselines.

- **CAA-AT + CutMix**—adversarial training based on constrained adaptive attacks, combined with sample mixing (Simonetto et al., 2024; Yun et al., 2019; Madry et al., 2018).

- **Drift-Resilient TabPFN**—in-context adaptation for temporal distribution shift (Helli et al., 2024).
- **FTAT**—fully test-time adaptation for tabular classification (Zhou et al., 2025).
- **DistPFN**—label-shift posterior correction applied to the TabICL-v2 base probabilities (Lee, 2026).

Regression-only baselines.

- **PGD-AT + Mixup**—projected-gradient adversarial training with sample mixing (Madry et al., 2018; Zhang et al., 2018).
- **DANN**—domain-adversarial representation alignment (Ganin et al., 2016).
- **SSA**—test-time subspace alignment for regression (Adachi et al., 2025).
- **V-REx**—risk extrapolation across training environments (Krueger et al., 2021).

B.4 Context Construction

This section specifies the context-construction interface, the registered constructors available to each variant, and the procedure used to select and aggregate their predictions. Each constructor maps a labeled source context and, when required, available unlabeled target features to a labeled context with the same feature interface as the target data.

B.4.1 Test-Time Target-Feature Interface and Singleton Calls

Each test-time call supplies one or more unlabeled target queries $X_t = \{x_i^t\}_{i=1}^{n_t}$. Source-only constructors use the labeled reference context and the current query. Target-dependent constructors additionally use the available target covariates required for target support, coordinate-wise quantiles, or location and scale. Target labels remain unavailable, and the frozen predictor and source-learned policy remain fixed. In the reported benchmarks, the held-out target partition for each evaluation condition is supplied through this interface.

For selection-shift constructions, the Base predictor first assigns pseudo-labels to target rows. When predicting x_i^t, the registered constructor may use a subset of the available peers $\{x_j^t : j \neq i\}$, with the queried row excluded. Source-only constructors use sampling, reweighting, and response adjustments derived from the labeled reference context.

For a singleton streaming request ($n_t = 1$), we disable constructors that require target-batch statistics or peer queries, including target-support matching, location and scale or quantile transport, and leave-one-out pseudo-labeling. The remaining source-only contexts use the ordinary fusion rule. If no valid source-only alternative is available, we set $\alpha_i = 0$ in Eq. (7) and return the Base prediction exactly.

B.4.2 Construction Ranges and Information Interface

Let (X_c, y_c) denote the labeled context available in an outer fold and let X_t denote the unlabeled target features available to a test-time call. Shared uses the five family-agnostic construction roles in Table 8 through a common input–output interface and reuses them across all change families and strengths. Prior-informed Specific retains this Shared registry and adds the constructors associated with a declared family z:

$$\mathcal{C}_{\text{Specific}}(z) = \mathcal{C}_{\text{Shared}} \cup \mathcal{C}_z. \tag{32}$$

The family determines candidate availability but does not provide the shift strength. Target labels are unavailable to every constructor. Feature projections are applied identically to X_c and X_t, and every classification subset retains all observed source classes.

Table 7: Information available during context construction and target inference.

Variant	Family identifier	Available data	Unavailable information
Shared	None	(X_c, y_c) and, when needed, X_t	Target labels and shift strength
Prior-informed Specific	Declared family z	(X_c, y_c) and, when needed, X_t	Target labels and shift strength

We use explicit percentile ranges throughout the registry. Source examples are first ordered by their robust distance to the target support and split into five equal-frequency intervals,

$$\mathcal{B}_t = \{[0, 20), [20, 40), [40, 60), [60, 80), [80, 100]\}. \tag{33}$$

The endpoints are empirical percentiles of target distance. The first interval contains the closest source examples. Successively broader contexts add the remaining intervals in order, with equal contributions from all included intervals.

For regression, the coarse response partition is

$$\mathcal{B}_y^{(4)} = \{[0, 25), [25, 50), [50, 75), [75, 100]\}, \tag{34}$$

and the fine partition is

$$\begin{aligned}\mathcal{B}_y^{(8)} = \{&[0, 12.5), [12.5, 25), [25, 37.5), [37.5, 50),\\ &[50, 62.5), [62.5, 75), [75, 87.5), [87.5, 100]\}.\end{aligned} \tag{35}$$

Balanced response contexts draw the same number of examples from every nonempty interval, using replacement when necessary. Symmetric response tails are written as

$$\mathcal{T}_y(q) = [0, q] \cup [100 - q, 100], \qquad q \in \{15, 20, 25, 30, 35\}\%, \tag{36}$$

where the registered candidates expand from narrow to broad tails. For classification, balancing is performed within each observed class; a binary balanced context allocates its rows as evenly as possible between the two classes.

B.4.3 Reusable Construction Operators

Target-support matching. We robustly standardize X_c and X_t using the source median and interquartile range. For source example x_j, let

$$d_t(x_j) = \frac{1}{|\mathcal{N}_t(x_j)|} \sum_{x \in \mathcal{N}_t(x_j)} \|\widetilde{x}_j - \widetilde{x}\|_2^2, \tag{37}$$

where $\mathcal{N}_t(x_j)$ is a small neighborhood in the available target set. Target-matched resampling uses

$$v_j = \frac{\exp[-d_t(x_j)/h]}{\sum_k \exp[-d_t(x_k)/h]}, \tag{38}$$

with h equal to the empirical median of positive distances. Sampling is balanced across the included intervals of $\mathcal{B}_t$; classification weights are additionally normalized within each class.

Local consistency and label repair. For classification, ten source neighbors define

$$c_j = \max_{a \in \mathcal{Y}} \frac{1}{|\mathcal{N}_c(x_j)|} \sum_{k \in \mathcal{N}_c(x_j)} \mathbb{I}(y_k = a). \tag{39}$$

Stable contexts retain high-consensus examples within every class. Label repair replaces y_j by the neighborhood majority at consensus levels of 60%, 70%, and 80%. A target-facing stable score, $s_j^{\text{stable}} = (1 - c_j) + 0.10 d_t(x_j)$, ranks examples within each class and target-support interval.

Representative support and filtering. Robust feature outlyingness is

$$u(x_j) = \frac{1}{d}\sum_{k=1}^{d} \min\left(\left|\frac{x_{jk} - \mathrm{med}(X_{c,k})}{\mathrm{IQR}(X_{c,k})}\right|, 10\right)^2. \tag{40}$$

Representative contexts rank examples by $s_j^{\mathrm{rep}} = d_t(x_j) + 0.35u(x_j)$ and sample low-score examples within every interval of $\mathcal{B}_t$. Classification samples within each class; regression samples within $\mathcal{B}_y^{(4)}$ or $\mathcal{B}_y^{(8)}$. Filtered contexts retain 70%, 75%, 80%, 85%, or 90% of the source rows, ranging from stronger to lighter filtering.

Feature views. Features are ranked by

$$s_k^{\mathrm{feat}} = |\operatorname{corr}(X_{c,k}, y_c)|. \tag{41}$$

Registered views remove the leading feature or a progressively larger leading set, retain the weakest 50% or 75% of features, symmetrically clip the outer 1%, 5%, or 10% of empirical feature values, or clip to $[m_k - cs_k, m_k + cs_k]$ with $c \in \{1.5, 2, 2.5, 3, 3.5, 4\}$, where m_k and s_k are the feature median and interquartile range. Repeated extreme levels may instead be replaced by m_k.

Response balancing and repair. Balanced regression contexts sample equally from the intervals of $\mathcal{B}_y^{(4)}$ or $\mathcal{B}_y^{(8)}$. Repeated responses in $\mathcal{T}_y(q)$ are reflected toward a robust center m:

$$y_j^{(\lambda)} = m - \lambda(y_j - m), \qquad \lambda \in \{20, 30, 40, 50, 60\}\%. \tag{42}$$

A second repair estimates m jointly from the lower and upper tail representatives. Global response contraction uses

$$y_j^{(\beta)} = m + \beta(y_j - m), \qquad \beta \in \{45, 60, 75, 90\}\%. \tag{43}$$

Source-to-target transport. With robust source and target moments (μ_s, σ_s) and (μ_t, σ_t), full location-scale transport is

$$T(X_c) = \frac{X_c - \mu_s}{\sigma_s}\sigma_t + \mu_t. \tag{44}$$

Partial transport uses

$$X_c^{(\alpha_{\mathrm{tr}})} = (1 - \alpha_{\mathrm{tr}})X_c + \alpha_{\mathrm{tr}}T(X_c), \qquad \alpha_{\mathrm{tr}} \in \{25, 50, 75, 100, 125, 150\}\%. \tag{45}$$

The first four levels range from partial to full alignment; the final two provide mild extrapolation when the target support extends beyond the source. Location-only transport preserves source scale, while quantile transport maps each empirical source quantile to its target counterpart. Transported rows can then be resampled within $\mathcal{B}_t$.

Support reconstruction and temporal filtering. Selection and Support constructors first merge feature-identical rows. They then rebuild support with four, eight, or sixteen target-quantile intervals, sampling equally from every nonempty interval. Deduplicated candidates retain between 80% and 100% of unique rows. Temporal constructors divide a cycle into 20 phase positions and filter 20%, 40%, 60%, 80%, or 100% of the registered affected phases. The retained rows may subsequently undergo transport and target-interval balancing.

B.4.4 Shared Constructor Registry

Table 8: Registered context constructions used by Shared.

Task	context	Construction
Classification	Target-neighbor	Standardize X_c and X_t using the source median and interquartile range, summarize the target support with five quantile anchors, and rank source examples by squared distance to the nearest anchor. Retain 80% of the closest examples while preserving every source class.
Classification	Representative	Rank source examples by the sum of target-support distance and 0.35 times robust feature outlyingness. Retain the lowest-scoring 70% while preserving every source class.

Task	context	Construction
Classification	Locally stable	Compute label disagreement over ten source neighbors and add 0.10 times the target-support distance. Retain the lowest-scoring 80% while preserving every source class.
Classification	Feature subspace	Rank features by their absolute source feature–label correlation and remove the leading predictive feature; all source rows and the remaining features are retained.
Classification	Low conflict	Rank examples by local label disagreement plus 0.20 times robust feature outlyingness. Retain the lowest-scoring 75% while preserving every source class.
Regression	Uniform bootstrap	Draw a full-size bootstrap from (X_c, y_c) with uniform probability and replacement.
Regression	Response balanced	Partition y_c into the eight equal-frequency intervals of $\mathcal{B}_y^{(8)}$ and draw the same number of examples from every nonempty interval, using replacement to preserve the source context size.
Regression	Target matched	Compute each source example's mean distance to its ten nearest target neighbors after robust standardization. Resample a full-size context with probabilities $v_j \propto \exp[-d_t(x_j)/h]$, where h is the median positive distance.
Regression	Distribution aligned	Map each source feature from its robust source location and scale to the corresponding target location and scale using Equation (44); retain y_c unchanged.
Regression	Aligned target matched	Apply full location–scale transport and then resample the transported rows with the target-matching probabilities used above.

Across LimiX-2M, TabPFN-v3, and TabDPT v1.2, Shared uses the common context-construction interface defined above. Registered views are constructed without shift-family or strength inputs, and the fitted policy is frozen before held-out inference. Held-out labels are used only to compute final metrics.

B.4.5 Prior-Informed Specific Constructor Registry

Specific retains all Shared contexts and activates the complete registered pool for family z. The pool has no protocol-level candidate-count cap. Each range, repair strength, subspace, and transport level below defines a separate candidate. This pool is the outer-fold selection space, not the deployed branch count: the selected member list Φ_z is frozen before held-out inference, which uses one Base pass and $R_z = |\Phi_z|$ alternative passes. For the latency experiment, Shared and Specific are both evaluated with five alternative contexts to provide a matched-cost comparison.

Controlled context perturbations.

Table 9: Family-directed Specific contexts under controlled source-context perturbations.

Family	Classification construction	Regression construction
Shortcut	Remove the leading or a progressively larger predictive feature set; retain the weakest 50% or 75% of features; clip the outer 1%, 5%, or 10% of the leading feature; and retain 70%–90% stable rows within each class.	Remove high shortcut-score features; retain weak $50\%/75\%$ subspaces; clip with $c \in \{1.5, 2, 2.5, 3, 3.5, 4\}$; and repair responses after equal sampling from $\mathcal{B}_y^{(4)}$.
Boundary	Repair labels using ten neighbors at 60%, 70%, and 80% consensus; select stable examples within each class and $\mathcal{B}_t$ interval; and repeat repair after removing the leading feature.	Detect paired responses in $\mathcal{T}_y(q)$ for $q \in \{15, 20, 25, 30, 35\}\%$; use $\lambda \in \{20, 30, 40, 50, 60\}\%$ reflection; and preserve equal coverage over $\mathcal{B}_y^{(4)}$.
AUC/ Rank-band	Split the leading-feature range into band and off-band regions; sample both regions within each class; repair labels at 60%, 70%, and 80% consensus; and include a clipped leading-feature view.	Detect repeated responses in $\mathcal{T}_y(q)$ for $q \in \{15, 20, 25, 30, 35\}\%$; repair with $\lambda \in \{20, 30, 40, 50, 60\}\%$; and combine the repair with feature clipping at $c \in \{1.5, 2, 2.5, 3, 3.5, 4\}$.
Local	Construct local contexts from the first one, two, and three cumulative intervals of $\mathcal{B}_t$; balance within each class; and apply neighborhood repair and conflict filtering.	Sample from cumulative intervals of $\mathcal{B}_t$ while preserving $\mathcal{B}_y^{(4)}$ coverage, then add tail-repaired and feature-clipped views.
Mixed	Combine all five target-support intervals, three consensus levels, $50\%/75\%$ weak-feature views, and 1%, 5%, or 10% feature-tail clipping.	Combine $\mathcal{B}_y^{(4)}$, $\mathcal{T}_y(q)$, $\lambda \in \{20, 30, 40, 50, 60\}\%$, $c \in \{1.5, 2, 2.5, 3, 3.5, 4\}$, and $50\%/75\%$ feature subspaces.

Distribution shifts.

Table 10: Family-directed Specific contexts under seven distribution-shift families.

Family	Classification construction	Regression construction
Covariate	Form square-root, linear, intermediate-power, and squared density-ratio weights; balance by class across all five $\mathcal{B}_t$ intervals; and add transport at $\alpha_{\mathrm{tr}} \in \{25, 50, 75, 100\}\%$.	Use location-scale or quantile transport at $\alpha_{\mathrm{tr}} \in \{25, 50, 75, 100, 125, 150\}\%$ and balance samples across $\mathcal{B}_t$ before and after transport.
Label	Interpolate from source to estimated target class proportions at 25%, 50%, 75%, and 100%; sample equally across observed classes; and include transported prior-matched contexts.	Sample equally from $\mathcal{B}_y^{(4)}$ and $\mathcal{B}_y^{(8)}$; contract responses with $\beta \in \{45, 60, 75, 90\}\%$; and clip responses in $\mathcal{T}_y(q)$ for $q \in \{15, 20, 25, 30, 35\}\%$.
Concept	Locate a changed boundary region using the recent labeled context; sample separately inside and outside it; and repair, filter, or downweight the region at 60%, 70%, and 80% consensus.	Remove 10%, 25%, 50%, 75%, or 100% of the strongest smooth feature-response component while preserving coverage over $\mathcal{B}_y^{(4)}$.
Selection	Balance source-target membership weights across all five $\mathcal{B}_t$ intervals and observed classes; add mixed contexts in which Base-pseudo-labeled target rows other than the current query occupy 50% or 75%.	Deduplicate rows; rebuild support with four, eight, and sixteen target quantile intervals; sample equally from each nonempty interval; and retain 80%–100% of unique rows.
Domain	Apply location, location-scale, and quantile transport at $\alpha_{\mathrm{tr}} \in \{25, 50, 75, 100\}\%$, then balance across the five target support intervals.	Apply location-scale and quantile transport at $\alpha_{\mathrm{tr}} \in \{25, 50, 75, 100, 125, 150\}\%$ and balance over $\mathcal{B}_t$ before and after transport.
Temporal	Set the historical share to 0%, 10%, 20%, 30%, 40%, or 50%, fill the remainder with recent rows, balance within each class, and add transported boundary-repair contexts.	Filter 20%, 40%, 60%, 80%, or 100% of registered phases in a 20-position cycle; transport the retained rows; and balance over target-quantile intervals.
Support	Move source support toward the target at $\alpha_{\mathrm{tr}} \in \{25, 50, 75, 100\}\%$; sample from cumulative intervals of $\mathcal{B}_t$; and maintain class balance.	Deduplicate rows; rebuild support with four, eight, and sixteen target quantile intervals; sample equally within each interval; and expand retained unique coverage from 80% to 100%.

Different ranges, repair strengths, subspaces, and transport levels are separate candidates. Target matching and transport use X_t but not target labels; no constructor receives manipulation strength.

B.4.6 Candidate Selection and Aggregation

Outer-fold policy learning. Let $\Theta = (\mathbf{u}, \mathbf{v}_Q, \mathbf{v}_S, a)$ for classification and $(\mathbf{u}, \mathbf{v}_S^{\mathrm{reg}}, t_y, a)$ for regression, with $\boldsymbol{\pi} = \mathrm{softmax}(\mathbf{u})$, $\boldsymbol{\eta}_Q = \mathrm{softplus}(\mathbf{v}_Q)$, $\boldsymbol{\eta}_S = \mathrm{softplus}(\mathbf{v}_S)$, $\boldsymbol{\eta}_S^{\mathrm{reg}} = \mathrm{softplus}(\mathbf{v}_S^{\mathrm{reg}})$, $\alpha_{\max} = \mathrm{sigmoid}(a)$, and $\tau_y = \mathrm{softplus}(t_y) + \epsilon_0$ for small $\epsilon_0 > 0$. For the current outer fold, P^{cal} is the distribution of episodes $e = (\mathcal{D}_{s,e}^{\mathrm{cal}}, \mathcal{X}_{t,e}^{\mathrm{cal}}, y_{t,e}^{\mathrm{cal}})$ formed from its training datasets. We fit

$$\Theta^* \in \arg\min_{\Theta} \mathbb{E}_{e \sim P^{\mathrm{cal}}} \left[\mathcal{L}_{\mathrm{task}} \left(o^{\mathrm{final}}(\mathcal{D}_{s,e}^{\mathrm{cal}}, \mathcal{X}_{t,e}^{\mathrm{cal}}; \Theta), y_{t,e}^{\mathrm{cal}} \right) \right]. \tag{46}$$

Here $y_{t,e}^{\mathrm{cal}}$ is used only by $\mathcal{L}_{\mathrm{task}}$ (cross-entropy for classification or squared error for regression), never by context construction. The resulting Θ^* and Φ are fixed before target inference, while the backbone parameters θ remain frozen.

Fusion policies are fitted by five-fold dataset-level out-of-fold evaluation. For each held-out dataset, aggregation parameters are fitted using the other outer folds and are then fixed for target inference. Calibration episodes span the fixed grid $\{0, 0.2, 0.5, 0.8, 1\}$ under the registered operators; calibration does not synthesize additional shifts. Shared pools families, whereas Specific fits one policy per declared family over a fixed constructor registry. Strength is not supplied to the deployed policy, and no held-out-dataset label enters fitting.

For a regression query x_i^t, the Base context produces $\hat{y}_i^{(0)}$, while the configurable alternatives produce the scalar prediction bank $\mathcal{P}_i = \{\hat{y}_i^{(r)}\}_{r=1}^{R}$. The interface does not assume a fixed list of context constructors; it only requires each branch to expose a scalar prediction. Equations (4) and (5) assign support-weighted member weights and form $\bar{y}_i$. Equation (6) then scores this supported aggregate with Base, and Eq. (7) performs the task-gated convex update. All aggregation and integration parameters are calibrated through Eq. (46) and fixed before target inference; the foundation model remains frozen.

Batch sharing and computational complexity. Let $\mathcal{B}$ and $\mathcal{Q}$ index the batch-shared and query-specific alternative branches, with $|\mathcal{B}| + |\mathcal{Q}| = R$. For a frozen TFM, a shared branch costs $T_{G_r}(n_s, n_t, d) + T_f(m_r, n_t)$, whereas a leave-one-out branch costs $\sum_i [T_{G_r}^{(i)}(n_s, n_t, d) + T_f(m_{ir}, 1)]$. Here T_G and T_f denote construction and forward costs, respectively; the Base branch adds $T_{G_0}(n_s, d) + T_f(m_0, n_t)$. Scoring and fusion across the R alternatives require $O(n_t RL)$ operations for classification and $O(n_t R)$ for regression.

B.5 Implementation Details

Table 11 summarizes the shared implementation settings; track-specific sample limits and context layouts are given above.

Table 11: Implementation settings for the public-data experiments.

Configuration	Setting
Backbone	TabICL 2.1.1 with `tabicl-classifier-v2-20260212.ckpt` and `tabicl-regressor-v2-20260212.ckpt`. Each context uses one estimator with batch size 8; backbone parameters remain frozen. Model inputs use `float32`, stored predictions and fusion use `float64`, and TabICL retains automatic mixed precision.
Data processing	Preprocessing is fitted on source data. Numerical values use median imputation. Classification categories are ordinally encoded, with unseen or missing values mapped to -1. Regression categories use most-frequent imputation and one-hot encoding; constant encoded columns are removed. Regression targets are standardized within the backbone.
context construction	Labeled-row budgets follow Table 4. Shared uses the fixed registry in Table 8. Specific retains Shared and uses the family-conditioned ranges in Tables 9 and 10, without a protocol-level cap on candidate-pool size.
Candidate selection	Shared retains its fixed five-context registry. For Specific, the complete family-conditioned registry defines the selection space. Five-fold dataset-level OOF selects Φ_z and fits source-side policy parameters on the other four folds; both are frozen for the held-out dataset. Selection uses cross-entropy for classification and squared error for regression.
Aggregation	Candidate predictions are combined using the query-dependent evidence weights in Eq. (5). The resulting aggregate is integrated with the Base prediction through Eqs. (6)–(7); its parameters are calibrated by Eq. (46).
Evaluation	Compared methods use identical source partitions, target batches, and dataset-level outer folds. Track-specific target-batch sizes are reported in Table 4.
Compute	Experiments run on an NVIDIA GeForce RTX 4090 with Python 3.12.3, PyTorch 2.8.0+cu128, scikit-learn 1.9.0, and NumPy 2.3.2.

C Backbone-Wise Strength Results

This section expands the backbone-wise macro-averages in Table 1 across shift strength. For TabICL-v2, the tables pair Accuracy/AUC and R^2/nMAE for Base, TabACE-Shared, and the prior-informed TabACE-Specific. Specific selects its frozen member list from the complete admissible pool without pre-truncating the selection space. Averaging the Mean rows over the four nonzero strengths recovers the corresponding R^2 entries in Table 1; nMAE supplies the complementary absolute-error view. Section C.1 reports the primary TabICL-v2 results, and Section C.2 evaluates the same interfaces on additional frozen backbones. Component ablations are reported separately in Appendix D.

C.1 TABICL-V2

Table 12: TabICL-v2 Accuracy/AUC across the five controlled Context perturbations. Entries are Accuracy/AUC; Mean gives the equal-weight family average.

Family	Configuration	Accuracy/AUC by strength				
		0	0.2	0.5	0.8	1.0
Shortcut	Base	0.804/0.901	0.699/0.842	0.537/0.724	0.380/0.578	0.274/0.464
	TabACE-shared	0.804/0.901	0.701/0.843	0.602/0.746	0.513/0.654	0.447/0.591
	TabACE-specific	0.804/0.900	0.713/0.842	0.627/0.761	0.557/0.685	0.513/0.635
Boundary	Base	0.804/0.901	0.711/0.856	0.558/0.761	0.399/0.615	0.305/0.495
	TabACE-shared	0.804/0.901	0.712/0.856	0.611/0.775	0.519/0.678	0.461/0.605
	TabACE-specific	0.804/0.900	0.723/0.857	0.627/0.788	0.561/0.713	0.516/0.668
AUC-band	Base	0.804/0.901	0.718/0.857	0.590/0.768	0.464/0.662	0.378/0.576
	TabACE-shared	0.804/0.901	0.718/0.857	0.620/0.773	0.545/0.695	0.489/0.634
	TabACE-specific	0.804/0.900	0.730/0.858	0.656/0.800	0.603/0.751	0.566/0.716
Local	Base	0.804/0.901	0.700/0.844	0.539/0.728	0.384/0.585	0.280/0.470
	TabACE-shared	0.804/0.901	0.701/0.844	0.605/0.750	0.514/0.657	0.445/0.587
	TabACE-specific	0.804/0.900	0.708/0.848	0.621/0.773	0.550/0.701	0.495/0.645
Mixed	Base	0.804/0.901	0.705/0.847	0.553/0.739	0.405/0.603	0.310/0.497
	TabACE-shared	0.804/0.901	0.706/0.848	0.607/0.755	0.518/0.663	0.461/0.597
	TabACE-specific	0.804/0.900	0.718/0.851	0.636/0.779	0.563/0.707	0.519/0.659
Mean	Base	0.804/0.901	0.707/0.849	0.555/0.744	0.406/0.609	0.309/0.500
	TabACE-shared	0.804/0.901	0.708/0.849	0.609/0.760	0.522/0.669	0.461/0.603
	TabACE-specific	0.804/0.900	0.718/0.851	0.633/0.780	0.567/0.711	0.522/0.664

Table 13: TabICL-v2 R^2/nMAE across the five controlled Context perturbations. Entries are R^2/nMAE; Mean gives the equal-weight family average.

Family	Configuration	R^2/nMAE by strength				
		0	0.2	0.5	0.8	1.0
Shortcut	Base	0.739/0.257	0.570/0.324	0.362/0.447	0.211/0.554	0.253/0.644
	TabACE-shared	0.738/0.260	0.570/0.324	0.362/0.447	0.212/0.554	0.307/0.625
	TabACE-specific	0.738/0.261	0.670/0.310	0.480/0.415	0.475/0.440	0.466/0.480
Boundary	Base	0.739/0.257	0.400/0.496	-0.941/0.994	-2.622/1.510	-4.942/2.055
	TabACE-shared	0.738/0.260	0.589/0.387	0.328/0.557	-0.007/0.748	0.201/0.670
	TabACE-specific	0.738/0.261	0.604/0.362	0.631/0.422	0.464/0.543	0.278/0.618
Rank-band	Base	0.739/0.257	0.072/0.544	-1.289/1.021	-3.059/1.581	-4.564/1.982
	TabACE-shared	0.738/0.260	0.558/0.384	0.358/0.504	0.096/0.671	0.194/0.673
	TabACE-specific	0.738/0.261	0.576/0.373	0.477/0.455	0.396/0.558	0.282/0.618
Local	Base	0.739/0.257	0.697/0.291	0.641/0.340	0.415/0.476	-0.965/1.077
	TabACE-shared	0.738/0.260	0.697/0.291	0.641/0.340	0.485/0.436	-0.074/0.810
	TabACE-specific	0.738/0.261	0.697/0.291	0.641/0.340	0.537/0.417	0.198/0.681
Mixed	Base	0.739/0.257	0.643/0.393	-0.012/0.796	-1.875/1.377	-5.530/2.139
	TabACE-shared	0.738/0.260	0.643/0.393	0.589/0.393	0.422/0.519	0.088/0.702
	TabACE-specific	0.738/0.261	0.673/0.346	0.570/0.406	0.459/0.506	0.130/0.673
Mean	Base	0.739/0.257	0.476/0.410	-0.248/0.720	-1.386/1.100	-3.149/1.579
	TabACE-shared	0.738/0.260	0.611/0.356	0.456/0.448	0.241/0.585	0.143/0.696
	TabACE-specific	0.738/0.261	0.644/0.336	0.560/0.408	0.466/0.493	0.271/0.614

Table 14: TabICL-v2 Accuracy/AUC across the seven distribution shifts. Entries are Accuracy/AUC; Mean gives the equal-weight family average.

Family	Configuration	Accuracy/AUC by strength				
		0	0.2	0.5	0.8	1.0
Covariate	Base	0.841/0.877	0.841/0.877	0.838/0.872	0.823/0.852	0.804/0.834
	TabACE-shared	0.841/0.875	0.846/0.880	0.845/0.877	0.825/0.857	0.804/0.836
	TabACE-specific	0.840/0.876	0.843/0.878	0.839/0.873	0.827/0.859	0.808/0.840
Label	Base	0.841/0.877	0.769/0.850	0.720/0.832	0.690/0.822	0.669/0.816
	TabACE-shared	0.841/0.875	0.769/0.850	0.730/0.832	0.690/0.822	0.669/0.816
	TabACE-specific	0.841/0.876	0.793/0.851	0.754/0.842	0.708/0.834	0.688/0.825
Concept	Base	0.841/0.877	0.788/0.819	0.711/0.766	0.643/0.718	0.594/0.681
	TabACE-shared	0.841/0.875	0.792/0.825	0.729/0.785	0.677/0.744	0.629/0.708
	TabACE-specific	0.841/0.876	0.802/0.852	0.786/0.846	0.790/0.848	0.786/0.835
Selection	Base	0.841/0.877	0.838/0.874	0.835/0.868	0.818/0.852	0.792/0.824
	TabACE-shared	0.841/0.875	0.844/0.875	0.839/0.869	0.824/0.853	0.800/0.830
	TabACE-specific	0.842/0.877	0.841/0.875	0.840/0.873	0.829/0.856	0.804/0.832
Domain	Base	0.841/0.877	0.789/0.862	0.775/0.849	0.761/0.842	0.751/0.838
	TabACE-shared	0.841/0.875	0.790/0.862	0.777/0.849	0.789/0.844	0.777/0.840
	TabACE-specific	0.840/0.875	0.835/0.873	0.834/0.871	0.835/0.872	0.834/0.871
Temporal	Base	0.841/0.877	0.805/0.838	0.769/0.813	0.731/0.785	0.720/0.780
	TabACE-shared	0.841/0.875	0.805/0.838	0.772/0.818	0.736/0.787	0.730/0.787
	TabACE-specific	0.846/0.879	0.806/0.838	0.795/0.840	0.770/0.824	0.773/0.830
Support	Base	0.841/0.877	0.807/0.861	0.774/0.849	0.764/0.836	0.758/0.819
	TabACE-shared	0.841/0.875	0.822/0.865	0.776/0.849	0.765/0.836	0.759/0.821
	TabACE-specific	0.837/0.876	0.833/0.870	0.830/0.865	0.830/0.866	0.831/0.864
Mean	Base	0.841/0.877	0.805/0.854	0.775/0.836	0.747/0.815	0.727/0.799
	TabACE-shared	0.841/0.875	0.810/0.856	0.781/0.840	0.758/0.821	0.738/0.806
	TabACE-specific	0.841/0.876	0.822/0.862	0.811/0.859	0.799/0.851	0.789/0.842

Table 15: TabICL-v2 R^2/nMAE across the seven distribution shifts. Entries are R^2/nMAE; Mean gives the equal-weight family average.

Family	Configuration	R^2/nMAE by strength				
		0	0.2	0.5	0.8	1.0
Covariate	Base	0.728/0.301	0.689/0.357	0.504/0.505	0.261/0.623	0.117/0.680
	TabACE-shared	0.728/0.304	0.691/0.355	0.567/0.456	0.492/0.497	0.411/0.545
	TabACE-specific	0.727/0.303	0.691/0.354	0.667/0.366	0.652/0.376	0.637/0.389
Label	Base	0.728/0.301	0.695/0.338	0.472/0.494	-0.035/0.713	-0.560/0.881
	TabACE-shared	0.728/0.304	0.701/0.335	0.497/0.483	0.021/0.696	-0.477/0.859
	TabACE-specific	0.727/0.303	0.698/0.335	0.725/0.308	0.685/0.348	0.593/0.420
Concept	Base	0.728/0.301	0.717/0.322	0.641/0.409	0.469/0.538	0.292/0.638
	TabACE-shared	0.728/0.304	0.717/0.321	0.642/0.409	0.473/0.537	0.298/0.637
	TabACE-specific	0.727/0.303	0.723/0.317	0.681/0.372	0.626/0.427	0.573/0.467
Selection	Base	0.728/0.301	0.724/0.305	0.724/0.307	0.707/0.324	0.674/0.343
	TabACE-shared	0.728/0.304	0.724/0.304	0.724/0.306	0.709/0.324	0.676/0.343
	TabACE-specific	0.727/0.303	0.726/0.304	0.727/0.305	0.711/0.323	0.680/0.341
Domain	Base	0.728/0.301	0.488/0.494	0.178/0.633	-0.035/0.718	-0.128/0.755
	TabACE-shared	0.728/0.304	0.615/0.418	0.581/0.436	0.511/0.477	0.477/0.491
	TabACE-specific	0.727/0.303	0.679/0.356	0.661/0.371	0.650/0.378	0.638/0.384
Temporal	Base	0.728/0.301	0.403/0.562	-0.178/0.807	-1.100/1.081	-1.841/1.272
	TabACE-shared	0.728/0.304	0.562/0.469	0.313/0.619	-0.148/0.819	-0.474/0.933
	TabACE-specific	0.727/0.303	0.581/0.442	0.552/0.467	0.509/0.499	0.491/0.516
Support	Base	0.728/0.301	0.725/0.305	0.712/0.314	0.683/0.332	0.646/0.356
	TabACE-shared	0.728/0.304	0.726/0.305	0.713/0.314	0.685/0.331	0.650/0.354
	TabACE-specific	0.727/0.303	0.727/0.305	0.716/0.313	0.690/0.329	0.658/0.351
Mean	Base	0.728/0.301	0.634/0.383	0.436/0.496	0.136/0.618	-0.114/0.704
	TabACE-shared	0.728/0.304	0.676/0.358	0.577/0.432	0.392/0.526	0.223/0.595
	TabACE-specific	0.727/0.303	0.689/0.345	0.676/0.357	0.646/0.383	0.610/0.410

C.2 Additional Backbones

For LimiX-2M, TabPFN-v3, and TabDPT v1.2, Figures 6, 7, and 8 show the same eight macro-average strength curves used in the main comparison. All three backbones use the Shared context-construction and evaluation interface in Appendix B.4, with fitted policies frozen before held-out inference.

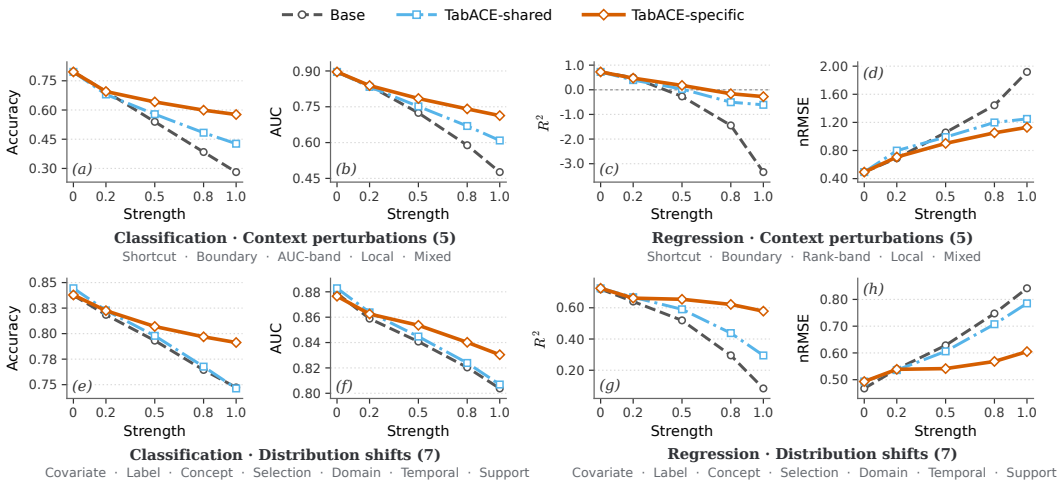


Figure 6: LimiX-2M results across shift strength. Rows aggregate the five context perturbations and seven distribution shifts; columns report Accuracy, AUC, R^2, and nRMSE (lower is better).

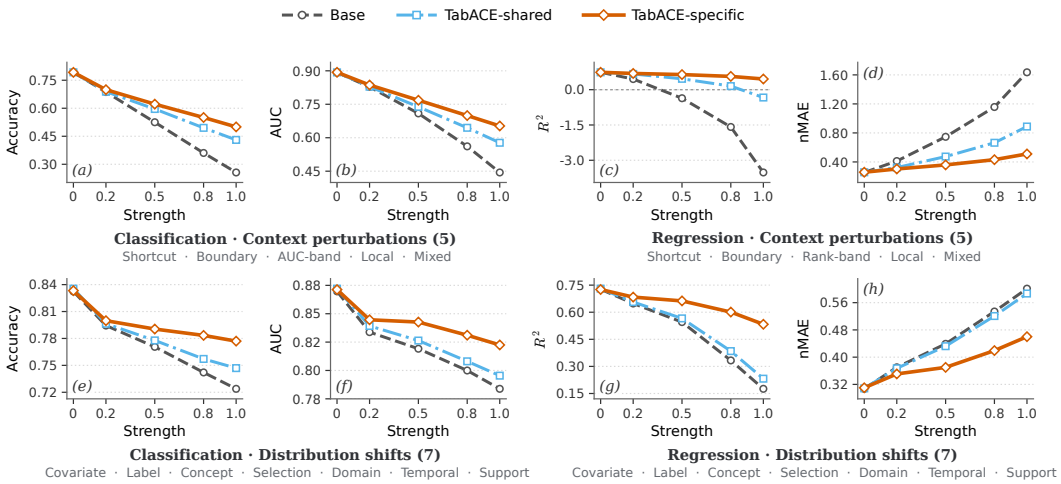


Figure 7: TabPFN-v3 results across shift strength. Rows aggregate the five context perturbations and seven distribution shifts; columns report Accuracy, AUC, R^2, and nMAE (lower is better).

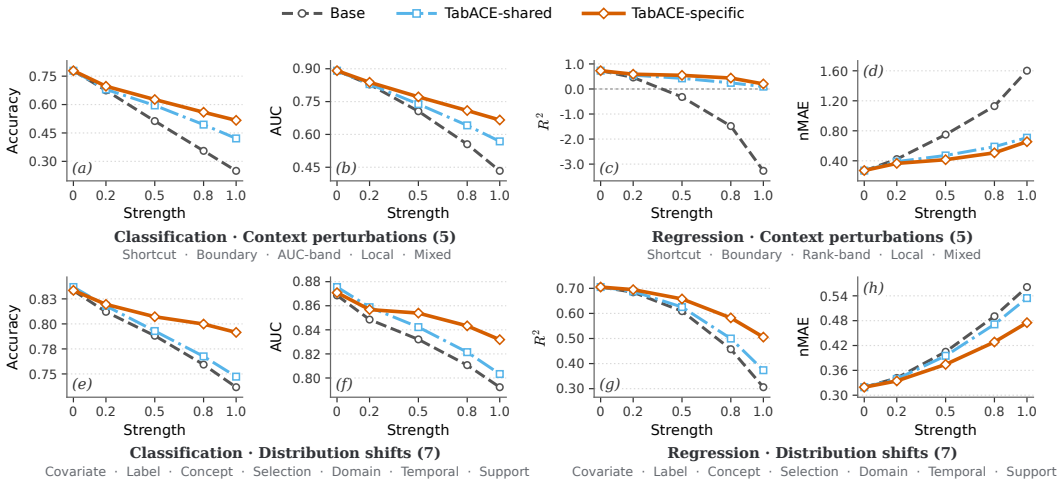


Figure 8: TabDPT v1.2 results across shift strength. Rows aggregate the five context perturbations and seven distribution shifts; columns report Accuracy, AUC, R^2, and nMAE (lower is better).

D Ablation Studies

The ablations use a five-context Shared reference bank selected on each outer-training fold. The context-budget study evaluates nested prefixes, while the remaining studies keep all five contexts fixed. This design isolates the effects of context count, aggregation, and formula components from context reselection.

D.1 Context Budget

We vary the context budget from $R = 1$ to $R = 5$ while holding construction and aggregation fixed. The single-context setting trails Base, after which all six metrics improve monotonically and $R = 5$ exceeds Base throughout. This trend is consistent with the complementarity analysis in Section 3.5: added contexts can cover previously unrepaired queries.

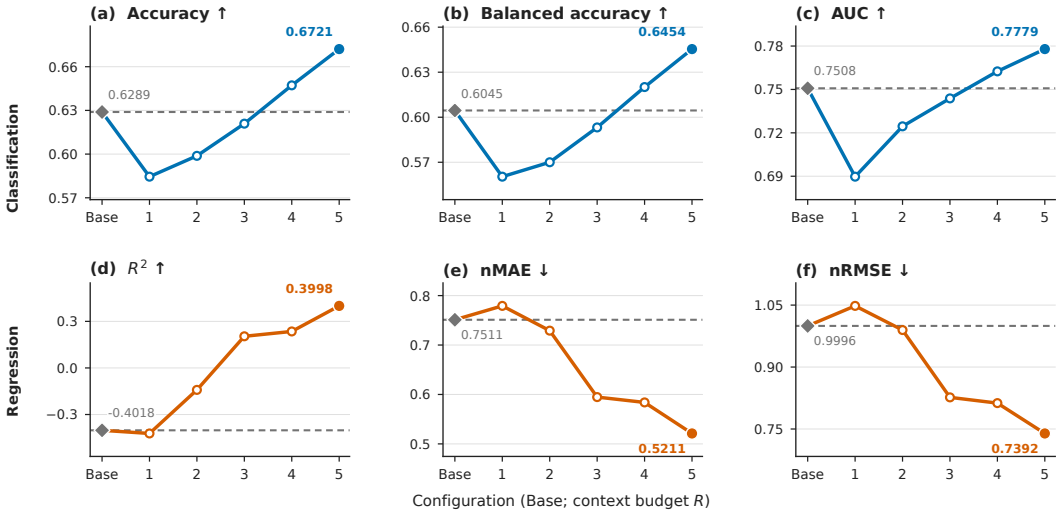
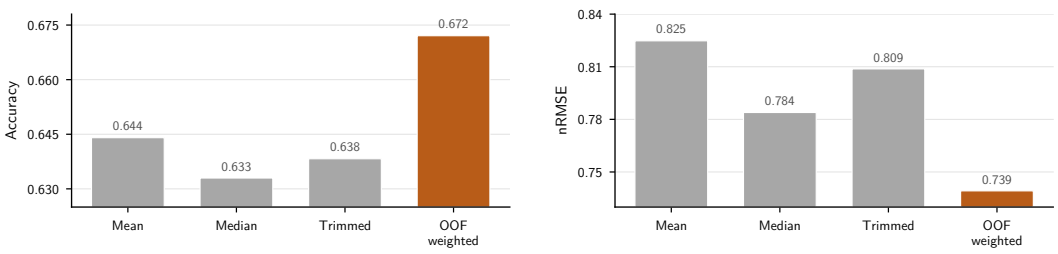


Figure 9: Context-count ablation over four nonzero strengths. Each curve starts from Base and continues through $R = 1, \ldots, 5$; dashed horizontal lines mark the Base level. The five-perturbation and seven-shift tracks receive equal macro weight. With only a few contexts, shift coverage is limited. Performance improves as complementary contexts are added, and $R = 5$ outperforms Base on all six metrics. Arrows indicate the preferred direction.

D.2 Aggregation Rule

Using the same five-context bank, we compare mean, median, trimmed-mean, and out-of-fold weighted aggregation while keeping context count and membership fixed. For classification, the coordinate-wise median and the mean after discarding the largest and smallest of five values per class are renormalized to the probability simplex; regression applies the same operators to scalar predictions. These context-only variants replace the entire aggregation rule, so none uses Q_i, S_i, or the Base gate. The out-of-fold weighted rule attains the highest classification Accuracy and lowest regression nRMSE, showing the benefit of source-learned reliability weights over unweighted statistical aggregation for this bank.



(*a*) Classification (*b*) Regression

Figure 10: Aggregation-rule ablation at $R = 5$, equally weighting the five-perturbation and seven-shift tracks. Panels compare classification Accuracy and regression nRMSE.

D.3 Fixed-Bank Component Ablation

Within this diagnostic configuration, we hold each fold-specific five-context bank fixed and remove one component at a time. All aggregation and gate parameters are selected on the corresponding outer-training folds, and outer-test labels are used only for evaluation. We evaluate five ablations: *w/o Confidence* sets $\eta_Q = 0$ while retaining Q_i in E_i; *w/o Agreement* sets $\eta_S = 0$; *Uniform Prior* sets $\pi_r = 1/R$; *w/o Base Gate* sets $U_i = 1$; *w/o context Gate* sets $E_i = 1$. *w/o Confidence* applies to classification, whose formula includes Q_i. The four nonzero strengths are averaged within the five-perturbation and seven-shift tracks before giving the tracks equal weight.

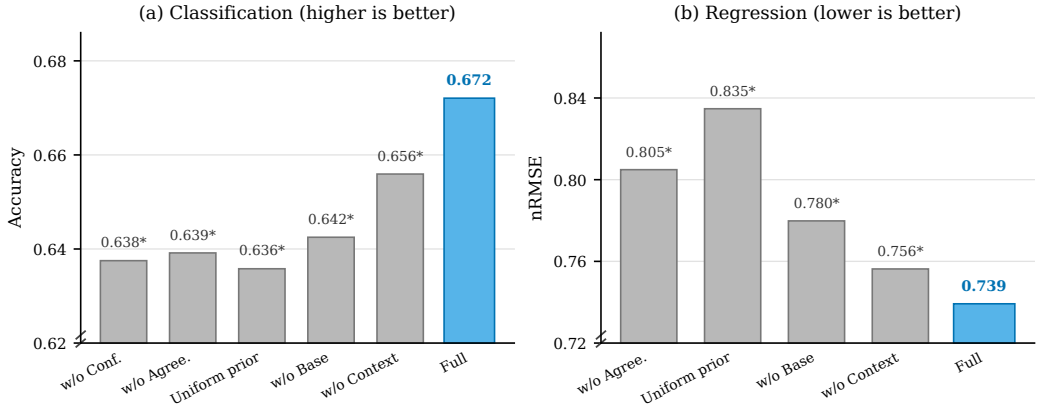


Figure 11: Component ablation for classification and regression. Higher Accuracy and lower nRMSE are better. Gray bars remove one component from the fixed-bank reference, shown by the final blue bar. Values are printed above each bar, and asterisks mark Holm-corrected paired sign-flip $p < 0.05$ relative to this reference. Break marks indicate truncated vertical axes.

The fixed-bank reference reaches 0.672 Accuracy and 0.739 nRMSE. Removing confidence, agreement, the learned prior, Base gating, or context gating reduces classification Accuracy to 0.638, 0.639, 0.636, 0.642, and 0.656, respectively. For regression, removing agreement, the learned prior, Base gating, or context gating increases nRMSE to 0.805, 0.835, 0.780, and 0.756. Every displayed difference is significant after Holm correction, with the largest degradation arising from the learned context prior on both tasks.

D.4 Gate Behavior across Evidence Bins

We partition 752,144 query predictions into task-wise tertiles of Base unreliability U_i and context evidence E_i. Each cell reports the mean interpolation weight and loss gain over Base, together with the fraction of material modifications ($\alpha_i \geq 0.5$) and their worsening rate.

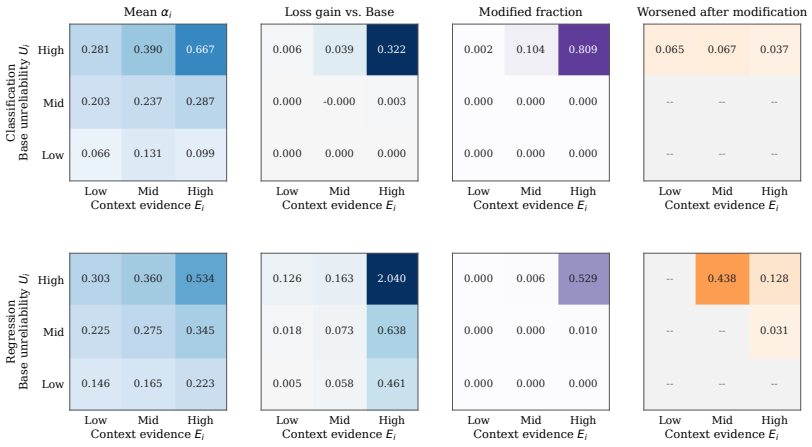


Figure 12: Gate behavior across Base-unreliability (U_i) and context-evidence (E_i) tertiles. Positive gain indicates improvement; gray cells contain no material modifications. In the high-U_i/high-E_i bin, classification/regression attain interpolation weights of 0.667/0.534, modification rates of 0.809/0.529, loss gains of 0.322/2.040, and worsening rates of 0.037/0.128; low/low loss gains are 0.000/0.005.

In the regression high-U_i/mid-E_i cell, only 0.6% of queries receive a material update; their conditional worsening rate is 43.8%, while the cell-level mean loss gain remains positive at 0.163.