PRIOR

# TabPFN-3: Technical Report

Prior Labs Team (see Appendix A for the list of contributors)

Tabular data underpins most high-value prediction problems in science and industry, and TabPFN has driven the foundation model revolution for this modality. Designed with feedback from our users, TabPFN-3 builds on this foundation to scale state-of-the-art performance to datasets with 1M training rows and substantially reduce training and inference time. Pretrained exclusively on synthetic data from our prior, TabPFN-3 dramatically pushes the frontier of tabular prediction and brings substantial gains on time series, relational, and tabular-text data.

**A new performance standard.** On the standard tabular benchmark TabArena, a forward pass of TabPFN-3 outperforms all other models, including tuned and ensembled baselines, by a significant margin, and pareto-dominates the speed/performance frontier. TabPFN-3 also scales to more diverse datasets: it ranks first on datasets with many classes, and beats 8-hour-tuned gradient-boosted-tree baselines on datasets up to 1M training rows and 200 features.

**Thinking mode.** TabPFN-3 introduces test-time compute scaling to tabular foundation models. Our API offering TabPFN-3-Plus (Thinking) exploits this to beat all non-TabPFN models by over 200 Elo on the standard TabArena benchmark, rising to 420 Elo on the largest data subset, and outperforming AutoGluon 1.5 extreme in less than a tenth of its runtime, without using LLMs, real data, internet search or any other model besides TabPFN.

**Broader capabilities.** TabPFN-3 extends the capabilities of our models, enabling SOTA prediction on many-class datasets, relational data (new SOTA foundation model on RelBenchV1) and tabular-text datasets (SOTA on TabSTAR via TabPFN-3-Plus). It also directly improves existing integrations of TabPFN: a specialized TabPFN-3 checkpoint, TabPFN-TS-3, ranks 2$^{nd}$ on the time-series benchmark fev-bench, and SHAP-value computation through `shapiq` is up to 120× faster with KV caching.

**An enterprise-ready model.** TabPFN-3 achieves this performance while being up to 20x faster than TabPFN-2.5. In addition, a reduced KV cache and row-chunking scale to 1M rows on a single H100 with fast inference speed.

We release TabPFN-3 under the `TABPFN-3.0 License v1.0`, permissive for research and internal evaluation. TabPFN-3-Plus (Thinking) is available via API and enterprise licensing including on-prem and VPC environments (AWS SageMaker, Azure AI Foundry).

DATE May 12, 2026

LICENSE `TABPFN-3.0 License v1.0` (see Section 5 for details)

DOCS https://docs.priorlabs.ai

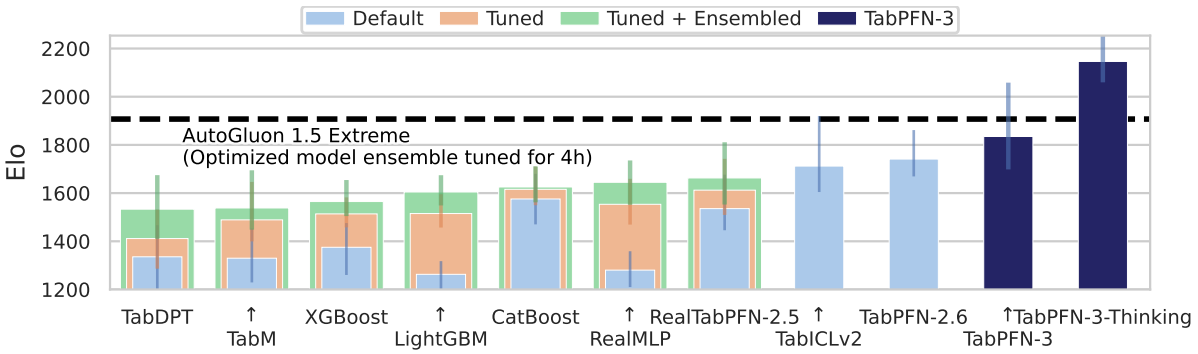



**Figure 1. Performance on the TabArena benchmark [1], largest data subset (10k-100k samples).** TabPFN-3 outperforms any other model in a forward pass. TabPFN-3-Plus (Thinking) is dramatically better yet, outperforming AutoGluon 1.5 extreme [2], a complex ensemble of models tuned for 4 hours, while being 10x faster.

# Contents

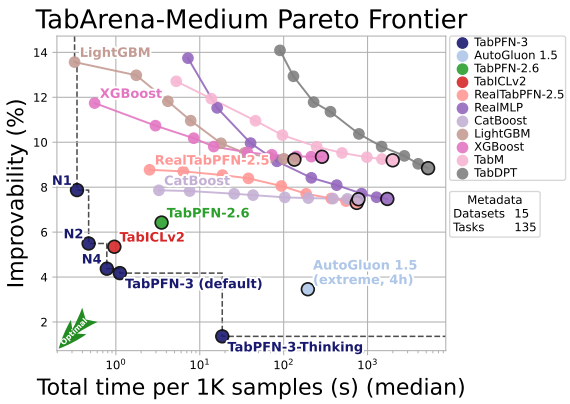



**Figure 2. TabPFN-3 dominates the Pareto frontier on the largest datasets in TabArena** (10k–100k rows). N1, N2, and N4 are model versions with 1, 2, and 4 estimators. Improvability measures how much worse a model is than the best per-dataset model. See Appendix E.2.1 and E.2.3 for details.

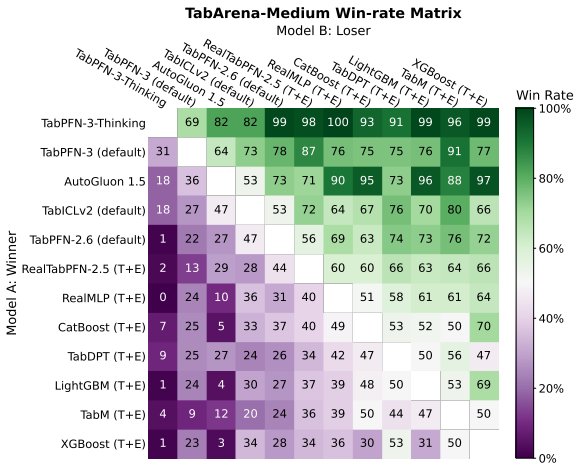



**Figure 3. Pairwise win rates on TabArena-medium** (10k–100k rows) for a curated set of the strongest models on TabArena. See Appendix E.2.4 for the full results.

# 1 Introduction

Tabular data sits at the core of operational decision-making across science and industry, including clinical risk prediction [3–5], credit scoring [6–8], predictive maintenance [9, 10], and scientific measurement [11, 12]. While gradient-boosted trees were the reliable default for decades [13–15], tabular foundation models have displaced them as the strongest predictors on standard small-to-medium-sized benchmarks over the last year [1].

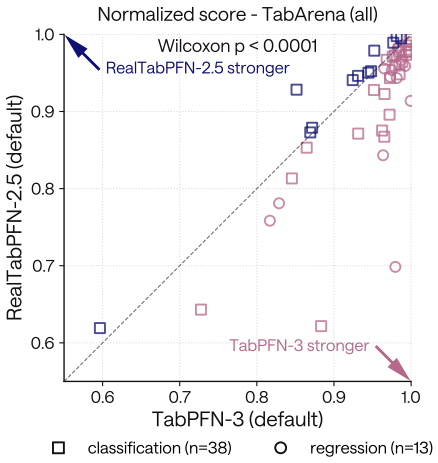
Earlier TabPFN releases established and extended this paradigm. TabPFN v1 [16] showed that a transformer pretrained on synthetic tasks could approximate Bayesian inference in a single forward pass, though only on a thousand rows of clean numerical data. TabPFN v2 [17] scaled this to 10,000 rows datasets with categorical features, missing values, and outliers, becoming the first tabular foundation model to outperform tuned gradient-boosted trees on standard benchmarks. TabPFN-2.5 [18] extended the strong performance to 100,000 rows and 2,000 features and matched four-hour-tuned ensembles in a single forward pass. Across these releases, an active research ecosystem of extensions grew on top of the core model – domains include time-series forecasting [19], causal inference [20–22], Bayesian optimization [23], graph learning [24, 25], interpretability [26, 27], reinforcement learning [28] – with over 200 published applications (see Appendix I) and more than three million PyPI downloads.

TabPFN-3 is shaped by the feedback from users and the entire ecosystem. To remove common bottlenecks, we scaled beyond a hundred thousand rows to one million rows, cut the memory and latency of inference at scale, added support for many-class classification, and honed our calibrated predictive distributions in a single forward pass. Furthermore, we carefully designed the TabPFN-3 model and training process to lift performance on both core tabular prediction as well as the many downstream extensions built on top of the open-source model, in particular time-series forecasting, multi-table relational data, and interpretability.

The remainder of this report describes the architecture, prior, and inference-time optimizations of TabPFN-3 (Section 2); evaluates its performance on public and internal benchmarks across classification, regression, many-class, time-series, and relational data (Section 3); surveys the adoption and ecosystem the model is built for (Section 4); and details licensing and availability (Section 5). Appendices provide architectural hyperparameters, prior visualizations, additional internal benchmarks, more detailed benchmark results and an extensive list of published TabPFN use cases. For installation and usage, see `https://docs.priorlabs.ai/`.

| Model | Rows | Features | Parameters | |
|---|---|---|---|---|
| | | | Clf. | Reg. |
| TabPFN-v1 | 1,000 | 100 | 26 M | — |
| TabPFN-v2 | 10,000 | 500 | 7 M | 11 M |
| TabPFN-2.5 | 100,000 | 2,000 | 11 M | 10 M |
| TabPFN-2.6 | 100,000 | 2,000 | 11 M | 13 M |
| TabPFN-3 | 1,000,000<br>100,000<br>1,000 | 200<br>2,000<br>20,000 | 53 M | 58 M |

**(a) Overview of previous TabPFN releases**, the maximal numbers of rows and features that they yielded state-of-the-art performance in, and their parameter counts. TabPFN-v1 supports classification datasets only.




**(b) TabPFN-3 presents a significant improvement against TabPFN-2.5**. We report the per-datasets scores on TabArena. The normalization procedure is described in Section F.1.

**Figure 4. Evolution and performance of the TabPFN model family.** The row and feature counts in the table denote benchmark-validated regimes where public and internal evaluations demonstrate state-of-the-art (SOTA) performance; larger or different row–feature configurations may be feasible, but are outside the validated SOTA envelope summarized here. Earlier releases are evaluated within a single row–feature regime, whereas TabPFN-3 is benchmarked along a cell-budget frontier: up to 1M rows at 200 features, 100k rows at 2,000 features, or 1k rows at 20,000 features. The right panel shows per-dataset scores across TabArena; points below the diagonal indicate stronger TabPFN-3 performance, with a Wilcoxon test confirming that the improvement is significant ($p < 0.0001$).

# 2 TabPFN-3

TabPFN-3 comes with a new architecture (Section 2.1), including an attention-based many-class decoder (Section 2.2), an improved preprocessing pipeline (Section 2.3), inference-time optimizations that enable scaling to one million rows on a single GPU (Section 2.4), and an improved synthetic SCM prior used for pre-training (Section 2.5). We also introduce the API and enterprise features TabPFN-3-Plus which handles text in tables natively and TabPFN-3-Plus (Thinking) which applies test-time-compute for dramatically improved performance (Section 2.6).

## 2.1 Architecture

An overview of TabPFN-3's full architecture is shown in Figure 5. TabPFN-3 introduces a substantially redesigned architecture that scales in-context learning to datasets with one million rows.

TabPFN v1 [16] used a transformer architecture to perform in-context learning (ICL) on embeddings of entire rows. TabPFN-2.x (v2, v2.5, v2.6) [17, 18] used a transformer architecture that alternates row-wise and feature-wise attention layers; this improves performance, but becomes prohibitively expensive as the dataset size grows. TabPFN-3 returns to TabPFN v1's ICL for embeddings of entire rows. It builds on the two-stage row-compression design introduced by Qu et al. [29, 30] in the TabICL architecture, which uses a column-wise feature embedding layer followed by row-wise feature aggregation to obtain the row representation that is used in a TabPFN v1-like ICL layer.

Before entering the two compression stages, we group features, similar to TabPFN-2.x, while adopting TabICLv2's[30] group assignment, which creates triplets by grouping each feature with two cyclically shifted neighbors. Each triplet is mapped to the hidden dimension of the model by a learned linear projection (cell embedding), and target-aware embeddings are added to the cell embeddings of training rows [30].

The resulting grouped feature embeddings are processed by the following three stages:

- **Stage 1: Feature distribution embedding (column-wise).** Each feature column is embedded independently using a transformer with an efficient inducing-point attention mechanism. This avoids the quadratic cost of full cross-row attention while still capturing column-level statistics at arbitrary dataset scales.
- **Stage 2: Feature aggregation (row-wise).** For each data point, a set of learned CLS tokens and the feature embeddings of that row attend to one another via non-causal attention, allowing cross-feature information to be distilled into a fixed number of vectors. Concatenating the CLS tokens' hidden states yields a single, fixed-dimensional embedding per row, decoupling the subsequent in-context learning stage from the number of input features.
- **Stage 3: In-context learning.** The row embeddings for the training and test sets are jointly passed to a transformer that performs in-context learning: training-row embeddings attend to one another to capture relationships within the training set, while test-row embeddings attend to training-row embeddings to produce predictions. Because each data point is now a single vector, this stage operates on a sequence proportional only to the number of rows, enabling efficient scaling to large datasets.

In Stages 1 and 3, and in the many-class decoder (introduced below), every attention layer applies the query-aware scalable softmax (QASSMax) [30], itself inspired from SSMAX [31], which rescales attention queries as a function of input length, improving length generalization of in-context learning to large training sets. Detailed architectural hyperparameters are provided in Appendix C.

TabPFN-3 introduces several architectural innovations on top of the three-stage architecture:

- **Attention-based many-class decoder.** For classification, the fixed-width MLP output head of previous TabPFN versions is replaced with an attention-based retrieval decoder that treats class prediction as soft nearest-neighbor retrieval over the in-context training set. The decoder is non-parametric in the class count, enabling native support for an arbitrary number of classes. A detailed description is given in Section 2.2.
- **Row-chunking.** A two-phase inference scheme that decouples peak GPU activation memory from dataset size (rows $\times$ columns), while producing outputs equivalent to the unchunked computation: we precompute the distribution embedder's inducing-vector summary once over the full training set, then stream rows through feature embedding and column aggregation in fixed-size chunks that reuse this cached summary as their attention key/value set. See Section 2.4.1 for more details.
- **Reduced KV cache via multi-query attention.** In the ICL transformer, test-row queries attend to train-row keys and values using a single KV head (multi-query attention), while train rows retain full multi-head attention. This allows reducing the per-estimator KV cache to approximately 7 GB for datasets of one million rows, enabling ultra-fast inference on common GPUs. This is described in detail in Section 2.4.2.
- **Orthogonal target embeddings.** Training labels are encoded with learned embeddings initialized via orthogonal decomposition, providing near-maximally separated class representations at the start of training and improving gradient flow in the many-class regime.
- **RMSNorm.** All normalization layers use RMSNorm in place of the layer normalization used in TabPFN-2.5. RMSNorm omits the mean-centering term, reducing compute while preserving training stability.
- **Native missing-value handling.** For each cell that is `NaN`, TabPFN-3 computes a binary indicator and concatenates it with the cell value before embedding. The model therefore receives an explicit signal about missing data and can condition its predictions accordingly, rather than relying on upstream imputation.

## 2.2 Many-class Decoder

For multiclass classification, TabPFN-3 replaces the fixed-width MLP classification head used in TabPFN-2.6 (and earlier versions) with an *attention-based retrieval decoder* over the in-context training set, which

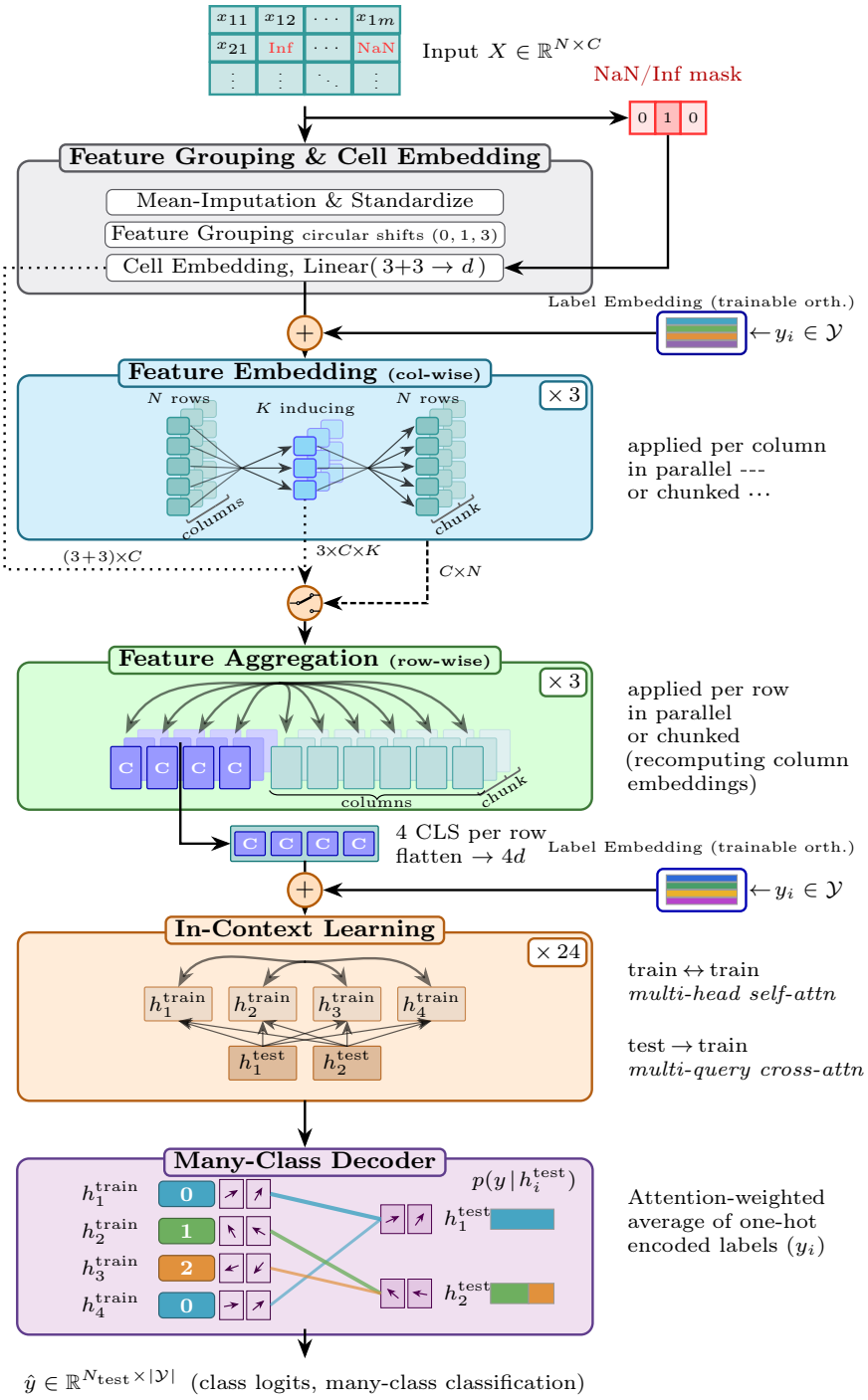



**Figure 5. Architecture of TabPFN-3, adapted from the TabICLv2 architecture.** Changes include adding novel orthogonal embeddings, the many class decoder, NaN/Inf indicator variables, and the option to use low-memory chunked inference (dotted paths; dashed paths signal fully parallel path). C refers to number of colums, N to the number of rows, K is the number of inducing points, and $\mathcal{Y}$ is the set of labels. Shown is TabPFN-3 for classification; the regression variant does not use the many-class decoder.

treats class prediction as a soft nearest-neighbor retrieval: the final-layer train embeddings $\{h_n^{\text{train}}\}_{n=1}^{N_{\text{train}}}$ act as keys, the corresponding one-hot label vectors $\mathbf{y}_n \in \{0,1\}^C$ as values, and test embeddings $h_m^{\text{test}}$ as queries. After the usual learned linear projections $W_Q, W_K$ and a multi-head split, the decoder computes

$$p_m \;=\; \frac{1}{H}\sum_{h=1}^{H}\sum_{n=1}^{N} \alpha_{m,n}^{(h)}\,\mathbf{y}_n, \qquad \alpha_{m,n}^{(h)} = \mathrm{softmax}_n\!\left(\frac{q_m^{(h)}\cdot k_n^{(h)}}{\sqrt{D_h}}\right),$$

that is: a (head-averaged) attention-weighted average of the in-context one-hot labels, which is then converted to logits via $\log\big(\mathrm{clip}(p_m)\big)$. This formulation has two consequences. First, classes are no longer tied to fixed output positions of a parametric head so the decoder is naturally permutation-equivariant in the class indices. Second, decoding is non-parametric in $C$: the decoder's parameters depend only on the embedding dimension and the number of attention heads, not on some $C_{\max}$, decoupling the head's capacity from the supported label cardinality.

**Class-count limit from pre-training.** Although the decoder is non-parametric in $C$, the trained TabPFN-3 still fixes a hard ceiling $C_{\max} = 160$ at pre-training time via three checkpoint-bound tensors: the trainable orthogonal label embeddings $E_{\mathrm{col}}, E_{\mathrm{icl}} \in \mathbb{R}^{C_{\max} \times D}$ used by the column encoder and the ICL transformer, and the one-hot value tensor consumed by the decoder. Enlarging $C_{\max}$ at pre-training therefore costs only $\mathcal{O}(C_{\max} D)$ extra parameters and no extra decode-time memory.

## 2.3 Preprocessing

As in previous versions, TabPFN-3 aggregates predictions across multiple estimators, each operating on a distinct combination of dataset permutations and feature transformations, forming an effective ensemble that enhances robustness and generalization. Individual estimators apply complementary feature transformations—combining robust scaling and soft clipping (following [32]) with quantile transformations and standard scaling—to balance stability and sensitivity across varying feature distributions. As in TabPFN-2.5, a subset of estimators augments the feature matrix with singular value decomposition (SVD) components, capturing high-energy directions of global variance.

TabPFN-3 introduces two further improvements to this pipeline. First, features are subsampled in a round-robin fashion, ensuring that each feature appears in at least one estimator and is never systematically excluded from the ensemble. For datasets exceeding 100,000 rows, random feature subsampling is replaced by an informed selection based on Gini importance derived from a lightweight tree model fitted on a subsample, focusing each estimator on the most discriminative features rather than an arbitrary subset. Second, feature transformations such as quantile normalization are now executed on GPU, substantially reducing preprocessing latency and making the pipeline practical at the larger dataset scales supported by TabPFN-3. As in TabPFN-2.5 [18], post-processing capabilities are available, including decision threshold tuning for metric-specific optimization (e.g., F1-score) and temperature scaling for probability calibration.

## 2.4 Inference Optimization

TabPFN-3 introduces several inference-time optimizations that together reduce its compute and memory footprint enough to scale to one-million-rows on a single GPU with sub-second inference latency.

### 2.4.1 Row-Chunking

TabPFN-3's pre-ICL stages—cell embedding, feature distribution embedding, and feature aggregation—materialize an $(n_{\mathrm{train}} + n_{\mathrm{test}}) \times n_{\mathrm{features}} \times d$ activation, so peak memory can saturate the GPU well before any operation becomes compute-bound. One solution is to offload activations to CPU memory or disk, as in TabICLv2 [30]. This however requires a large amount of CPU memory (250GB for a $1\mathrm{M} \times 500$ table in Qu et al. [30]), or otherwise incurs substantial I/O overhead (Qu et al. [30] report a 4x slowdown). We instead stream the row dimension in fixed-size slices and keep all activations on the GPU.

A naive row-wise stream is not directly applicable: the distribution embedder summarizes the training set into a fixed-size[1] set of inducing points via cross-attention over all training rows, and splitting that call across chunks would change its semantics. TabPFN-3 resolves this with a two-phase scheme exactly equivalent to the unchunked computation: (i) the inducing states are computed once over the full training set, chunked along the (independent) column dimension to bound its own memory cost; (ii) rows are then streamed through feature distribution embedding and the feature aggregator in fixed-size chunks, each reusing the precomputed inducing states as its attention key/value set, and the per-chunk row embeddings are concatenated along the row axis. The scheme adds a small overhead from recomputing cell embeddings in phase (ii) but avoids the disk-bandwidth bottleneck. We enable chunking when $n_{\mathrm{train}} + n_{\mathrm{test}} > 2048$.

[1] We use 128 inducing points, much smaller than the dataset sizes of interest, which often exceed 100,000 rows.

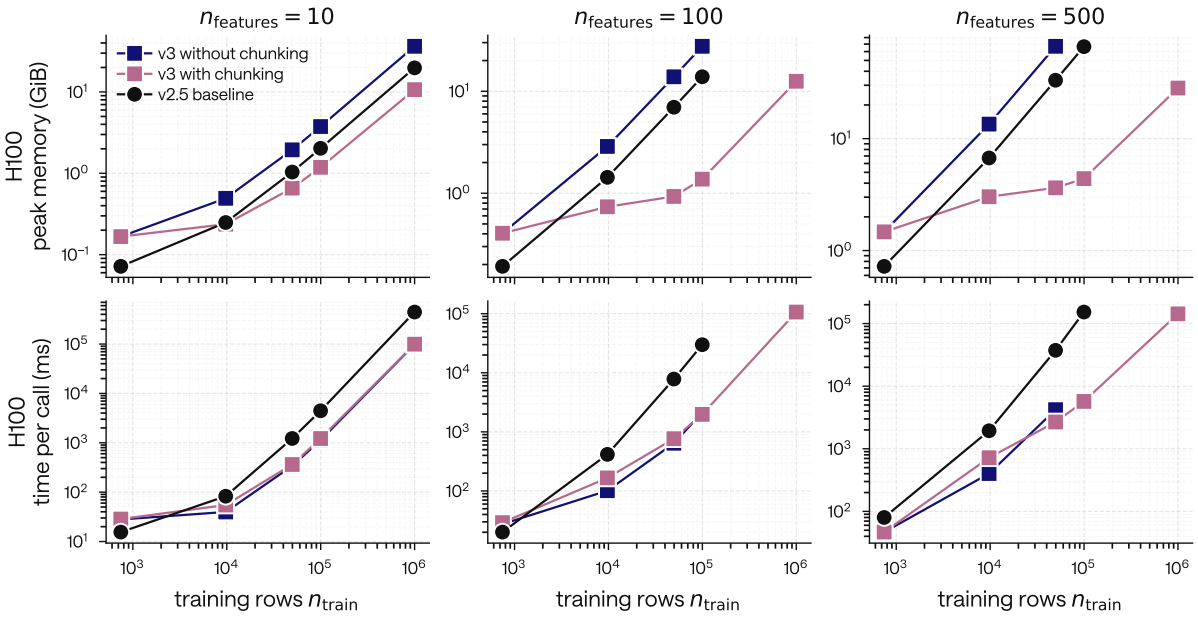


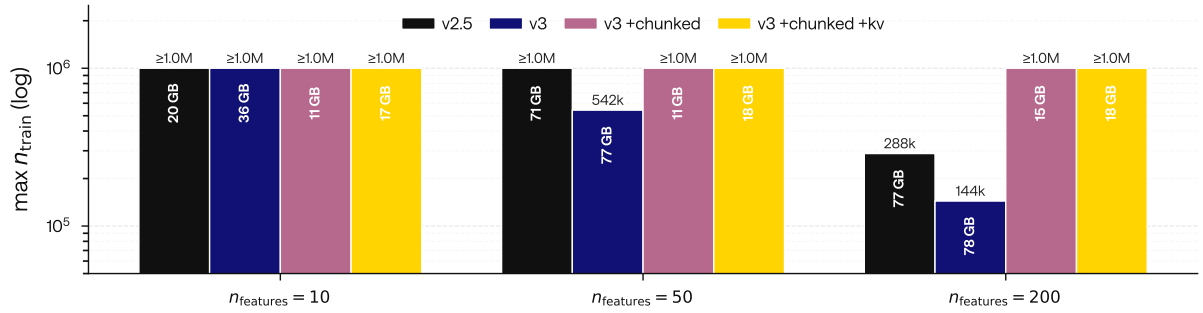

**Figure 6. Chunking flattens the peak-memory without impacting the time-per-call.** Model forward pass without preprocessing, measured on a H100, for $n_{\text{features}} \in \{10, 100, 500\}$. Top row: peak GPU memory (GiB) versus number of training rows; bottom row: time per call (ms). Three series per panel: TabPFN-3 without chunking (blue), TabPFN-3 with chunking (pink), and the TabPFN-2.5 baseline (black). Both axes are log-scaled. Note that *TabPFN-3 is much faster than TabPFN-2.5, especially at large feature counts.*

Figure 6 highlights the different memory–compute trade-offs of TabPFN-3 and TabPFN-2.5. Without chunking, the peak memory of TabPFN-3 grows steeply with $n_{\text{train}}$ and $n_{\text{features}}$. This is because the model carries a pre-ICL activation $n_{\text{features}}$-wide through cell embedding, feature distribution embedding, and feature aggregation before collapsing the feature axis into a single row representation for the ICL transformer. By contrast, TabPFN-2.5 alternates row- and column-attention layers over a representation grouped into $n_{\text{features}}/3$ tokens, and therefore never materialises a tensor wider than this. This explains why TabPFN-3's unchunked peak memory exceeds TabPFN-2.5's. Applying row-chunking to TabPFN-3 flattens peak memory with respect to $n_{\text{features}}$ and yields an approximately $\sim 5\times$ reduction at the largest shapes, enabling 1M-row inference, while incurring only a small wall-clock overhead of a few percent near $n_{\text{train}} \approx 10^4$ that becomes amortised at larger scales once the $n_{\text{train}}^2$ ICL row-attention dominates. At the same time, the feature-collapsed row representation gives TabPFN-3 a substantial runtime advantage at large $n_{\text{train}}$ or $n_{\text{features}}$ since its ICL row-attention scales as $n_{\text{train}}^2$ independently of $n_{\text{features}}$, whereas TabPFN-2.5's row attention retains linear dependence on $n_{\text{features}}$ and scales with $n_{\text{features}} \cdot n_{\text{train}}^2$.

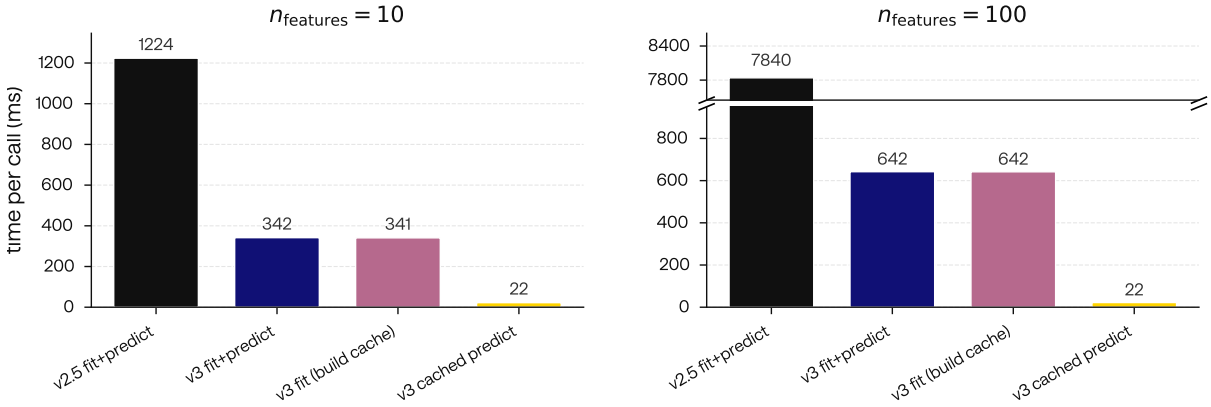
### 2.4.2 Fast Inference with a Small KV-cache

Being an in-context-learning model, TabPFN-3 combines training (fit) and inference (predict) in one forward pass. While this allows for very fast training, it can make online or batched predictions too slow for production usecases. Caching the keys and values (KVs) from the train set removes this issue. While KV-caching has been available in our previous models, the memory cost of the cache was prohibitive for larger datasets. TabPFN-3 solves this in two ways:

- Compared to TabPFN-2.5, which needs to store an embedding for each cell of the table, TabPFN-3 only needs to store three components: the per-block inducing states produced by the feature distribution embedder, the train-side keys and values of the ICL self-attention at every transformer block in the ICL stage; as well as the train embeddings of the final ICL layer, which are consumed by the many-class decoder. The inducing states are small and the two other components only scale with the number of rows rather than rows $\times$ features.
- We use multi-query with only a single head for cross attention between test and train samples, reducing KV-cache size by a factor of eight.




**(a) Chunking eliminates the OOM frontier; the KV-cache adds memory essentially constant in $n_{\text{feature}}$.** Maximum $n_{\text{train}}$ that fits on one 80 GiB H100 for $n_{\text{features}} \in \{10, 50, 200\}$. Bars: TabPFN-2.5, TabPFN-3, TabPFN-3 + chunking, TabPFN-3 + chunking + KV-cache. White labels: peak memory; "$\geq$ 1.0M" marks bars that hit the search cap.




**(b) Cached predict is 1–2 orders of magnitude faster than the uncached TabPFN-3.** Time per model forward pass without preprocessing on H100 at $n_{\text{train}} = 50{,}000$, $n_{\text{test}} = 100$, $n_{\text{features}} \in \{10, 100\}$. Bars: TabPFN-2.5 cold fit+predict, TabPFN-3 cold fit+predict, TabPFN-3 fit-with-cache, TabPFN-3 cached predict.

**Figure 7.** KV-cache on H100 for a single estimator without preprocessing: OOM frontier with chunking and KV-cache (a) and cached-predict latency vs. uncached paths (b).

This achieves a KV-cache size of 7GiB per estimator for 1M rows datasets, making TabPFN-3's default 8 estimators usable on common GPUs even for the largest datasets we support. As can be seen in Figure 7a, peak memory of (chunked) cache-predict is basically flat across feature sizes. On an H100, cached-predict is one to three orders of magnitude faster than either the TabPFN-2.5 baseline or TabPFN-3's own cold "fit+predict" path (Figure 7b), achieving between 0.1 and 3 ms/test point for batches of 100 test points. The fit-with-cache call costs essentially the same as the cold fit+predict at every measured shape, including $n_{\text{train}} = 10^6$ where both complete in ∼107 s (Figure 8).

### 2.4.3 Model Distillation

In production environments constrained by latency or memory budgets, hardware availability, or regulatory requirements that mandate familiar model classes, TabPFN-3 also supports distillation into dataset-specific MLPs or tree ensembles via the engine introduced with TabPFN-2.5 [18]. The distilled artifact runs on CPU at the sub-millisecond latency of a standard MLP or tree ensemble while retaining most of TabPFN-3's predictive performance on the dataset it was distilled for.

### 2.4.4 Compilation and FlashAttention-3

TabPFN-3 ships with two opt-in performance features that target different bottlenecks: `torch.compile`, which fuses dispatch on the non-attention hot paths, and FlashAttention-3 (FA3) [33], a Hopper-specific kernel for the in-context-learning attention. On MI-250x, `torch.compile` reaches up to 1.58× speedup on the non-chunked forward pass; on H100, FA3 reaches 1.5–1.7× at $n_{\text{train}} = 10^6$ over the SDPA fallback. Both compose cleanly with row chunking and are auto-detected at runtime; see Appendix G.1 for the full

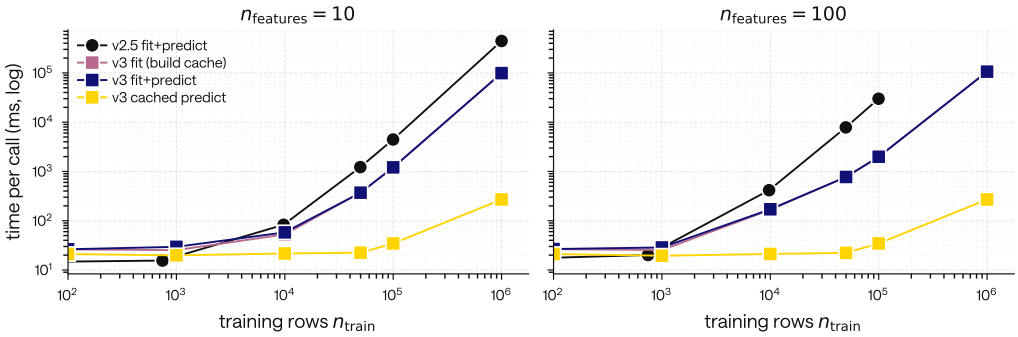



**Figure 8. TabPFN-3's KV-cached predict allows for one to three orders of magnitude speedup**. We report results for a single estimator without preprocessing on an H100, for $n_{\text{features}} \in \{10, 100\}$ and $n_{\text{test}} = 100$. Four series per panel: TabPFN-2.5 `fit+predict` (black, baseline), TabPFN-3 cold `fit+predict` (blue, no cache reuse), TabPFN-3 `fit (build cache)` that builds the cache (magenta – overlaps the cold curve since the train-side work is identical, the cache is just retained), and TabPFN-3 cached `predict` (yellow). The KV-Cache is built under the deployed multi-query test-side configuration ($n_{\text{kv,test}} = 1$).

measurements and per-shape breakdowns.

### 2.4.5 Improved interpretability for TabPFN

TabPFN-3's reduced KV-cache (Section 2.4.2) and fast inference make interpretability extensions significantly more practical.

Through the `tabpfn-extensions` package, TabPFN is directly integrated with the popular `shapiq` library [34], enabling efficient approximation of any-order Shapley interactions. Figure 38 in the Appendix shows both the absolute runtime and the relative speed-ups achieved by KV caching. For large datasets, KV caching provides more than 120× efficiency gains, reducing the runtime per test row to 1.08 seconds even for a training table with 200k rows and 500 features.

## 2.5 Synthetic Prior

Following previous TabPFN model variants [16–18], TabPFN-3 is trained on synthetically generated data based on our Structural Causal Model (SCM) prior. A schematic flow chart demonstrating how our SCM prior works is shown in Figure 9.

Our philosophy in designing our prior is to maximize breadth of possible datasets while capturing the structure models will encounter in real-world data. The result is an updated, more sophisticated prior that allows us to scale up training and continue extracting signal from the wide range of synthetic datasets it generates: our final TabPFN-3 model was trained on more than 8 trillion tokens.

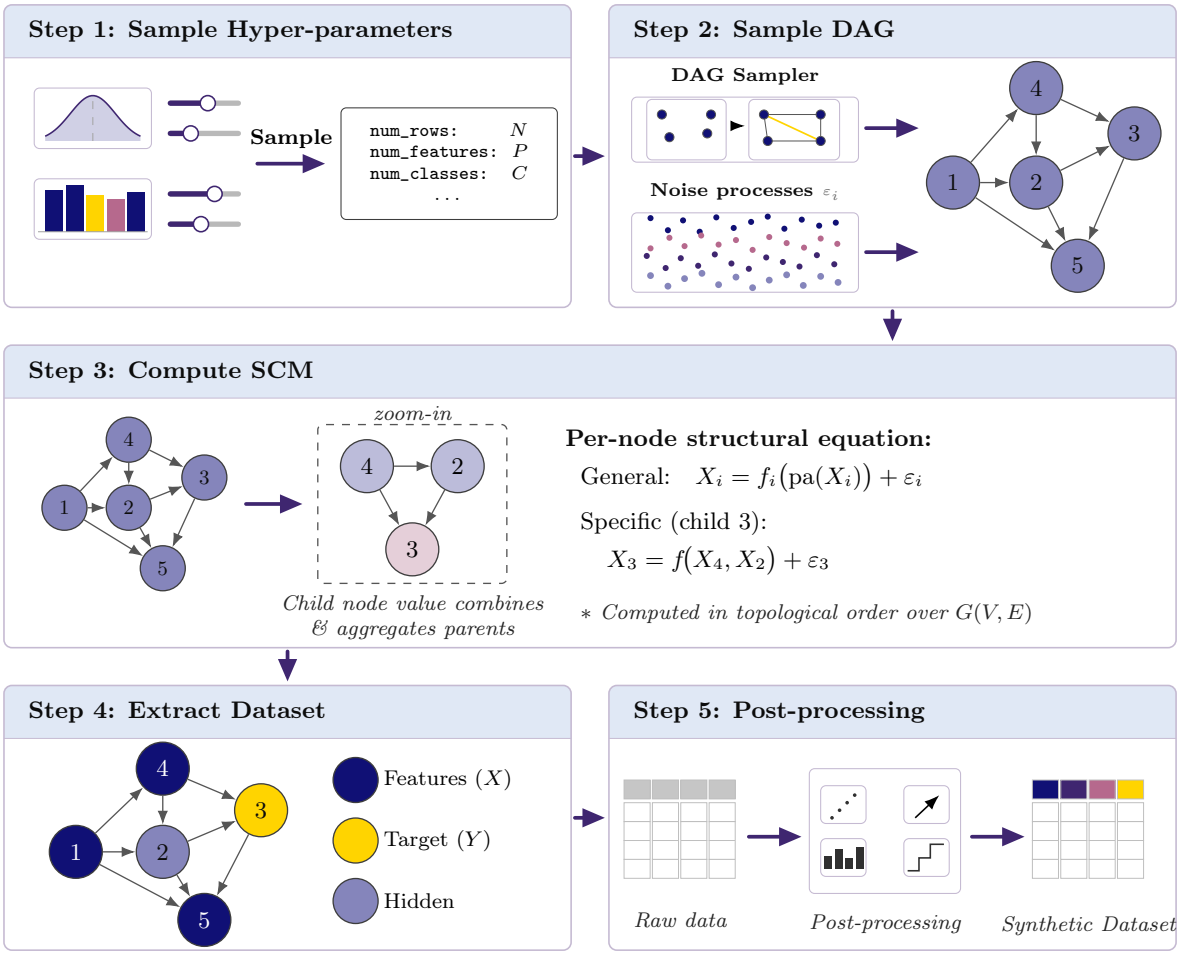



**Figure 9. Schematic visualization of our SCM prior.** (i) We first sample high-level hyperparameters for the dataset, including number of features and number of rows. (ii) Based on the hyperparameters, we utilize our graph sampling algorithms to generate a directed acyclic graph (DAG) underlying our SCM; in parallel, an i.i.d. noise sample $\varepsilon_i$ is drawn per node (each colour shade in the lower-left mini-panel corresponds to a different node). (iii) We compute a topological ordering of the DAG. Based on this, we create a computational graph: First we fill root nodes (i.e. exogenous variables) and subsequently traverse the computational graph in topological order, combining parent nodes using our combiner mechanisms and activations to propagate values to the child nodes. (iv) We choose suitable features and target variables from our fully computed SCM. (v) We apply post-processing to the dataset.

1. **Graph generation.** We expand the distribution of graphs underlying the SCM by introducing new sampling algorithms, enabling richer structural diversity. Sample graphs are shown in Figure 23.

2. **Combiner mechanisms.** We introduce a host of new combiner mechanisms that combine values of parent nodes to propagate values to child nodes, some examples of which are visualized for a simple two-dimensional case in Figure 24. Increasing the variety of functional forms by which child nodes depend on the respective parent nodes allows for richer node relationships in the SCM.

3. **Categorical variables.** Compared to TabPFN-2.5, we reworked the treatment of categorical variables in our SCM, moving from a comparatively simple categorical data model to more expressive variants.

4. **High-frequency oscillators.** TabPFN-2.5 struggled with high-frequency oscillations despite performing well on sinusoidal data generally. Improved sinusoidal activations give TabPFN-3 strong performance across the full frequency spectrum.

5. **Spatial prior.** Many tabular datasets have underlying spatial structure (e.g. datasets containing longitude and latitude as covariates, grids of sensors, etc.). We add spatial activations that allow our prior to encode spatial relationships between variables.

6. **Many-class prior.** The flexible many-class decoder in TabPFN-3 enables native classification support for an arbitrary number of classes. We match this architectural design in the prior, ensuring high quality datasets that enable state-of-the-art downstream performance from binary datasets to datasets with hundreds of classes.

7. **Temporal prior.** Many tabular datasets have temporal structure: rows are collected over intervals of time, train and test splits are often ordered by time rather than drawn i.i.d., and temporal dependencies between variables are common. We extend the SCM into a discrete-time Dynamic Structural Causal Model [35].

8. **Out-of-distribution prior.** We add out-of-distribution prediction tasks, allowing models trained on our prior data to remain performant under distribution shifts, as well as moving from pure interpolation to extrapolation. A simple example highlighting how our o.o.d. prior allows TabPFN-3 to perform extrapolation is shown in Figure 26 - a capability that is notably absent from most tree-based algorithms as well as most other tabular foundation models.

### 2.6 TabPFN-3-Plus and Thinking mode

On top of TabPFN-3, which we release open-source, our API and enterprise deployments provide access to TabPFN-3-Plus and its thinking mode "TabPFN-3-Plus (Thinking)" (named TabPFN-3-Thinking in our plots) These variants are fully compatible with the open-source TabPFN-3 interface and can be used as a drop-in replacement, while offering additional capabilities:

**Native text-feature support.** TabPFN-3-Plus accepts string-valued columns directly, without requiring upstream featurization. Free-text fields – such as product names, insurance claim descriptions, or customer reviews – are encoded jointly with numeric and categorical features inside the model, so cross-feature interactions between text and structured columns are learned end-to-end rather than imposed by a fixed encoder.

**Thinking mode.** TabPFN-3-Plus (Thinking) applies additional inference-time computation on top of TabPFN-3-Plus to push prediction quality further. Thinking mode composes with native text-feature support, so a single call can handle mixed numerical, categorical, and text columns under the same inference-time-compute regime. We emphasize that our Thinking mode achieves this strong performance while only relying on TabPFN, without using LLMs, real data, internet search, or any other model.

TabPFN-3-Plus, including Thinking mode, is available through our API and through enterprise deployments including on-prem and VPC deployment on AWS SageMaker and Azure AI Foundry; see Section 5 for licensing and access. Benchmark results are reported in Sections 3.1.1 (TabArena), 3.1.3 (TabSTAR), and 3.2.1 (Large data).

## 3 Experimental Results

In this section, we report experimental results across a variety of benchmarks. In Section 3.1, we focus on public tabular benchmarks: TabArena [1], TALENT [36], and the text-tabular TabSTAR collection [37]. Section 3.2 describes internal benchmarks spanning various subtypes of tabular learning, including large-scale datasets and features, many-class classification, and quantile regression. The subsequent sections extend beyond classic tabular learning: Section 3.3 addresses time-series data, Section 3.4 covers relational learning, and Section 3.6 focuses on embeddings.

### 3.1 Public Tabular Benchmarks

#### 3.1.1 TabArena

TabArena [1] (NeurIPS 2025 Datasets & Benchmarks) is a recent and heavily curated tabular benchmark, based on the largest number of candidate datasets considered, and created and maintained by open-source contributors from a wide range of institutions. In particular, it compares a large and regularly updated

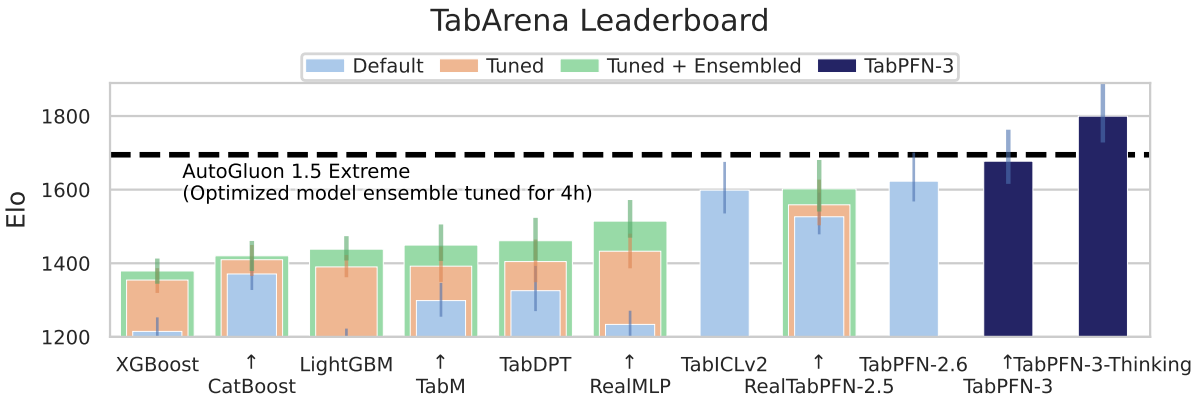



**Figure 10. TabPFN-3 performance on the standard TabArena benchmark** [1], including all 51 datasets (up to 100K rows). TabPFN-3 outperforms any other model in a forward pass, while TabPFN-3-Plus (Thinking) strongly outperforms all existing methods, including AutoGluon 1.5 extreme [2], a complex ensemble of models including TabPFN v2 tuned for 4 hours, in less than a tenth of the runtime.

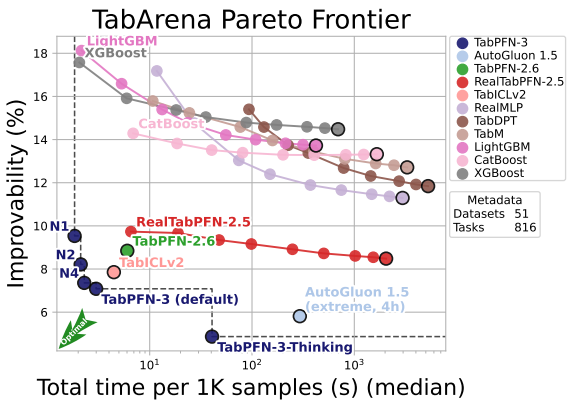



**Figure 11. Pareto frontier on TabArena**: trade-off between prediction quality and total training + inference cost. N1, N2, and N4 are TabPFN-3 versions with 1, 2, and 4 estimators. Improvability measures how much a model would improve by switching to the best model on each individual dataset, see Appendix E.2.1.

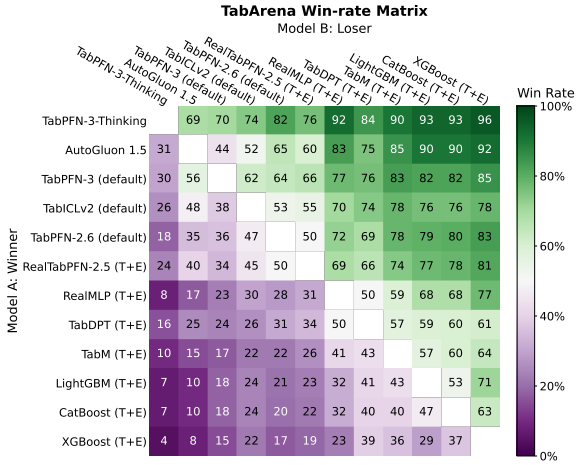



**Figure 12. Pairwise win rates on TabArena** for a curated set of the strongest models on TabArena. See Appendix E.2.4 for the full results.

list of recent models, including tree-based models like CatBoost [38], LightGBM [39] or XGBoost [40], as well as newer deep-learning models like RealMLP [32], TabM [41], ModernNCA [42] or xRFM [43], the AutoML system AutoGluon [2], and other Tabular Foundation Models like TabICL [29, 30], TabDPT [44], TabSTAR [37], LimiX [45], Mitra [46] or TabPFN v2 [17]. The benchmark contains a set of 51 datasets selected from 1053 to be representative of real-world tabular data. See Erickson et al. [1] for the list of datasets and Section E.2.1 for definitions of TabArena's Elo and Improvability metrics.

**Pushing the performance frontier on TabArena.** Figure 10 shows the performance of TabPFN-3 and TabPFN-3-Plus (Thinking) on TabArena. TabPFN-3 outperforms in one forward pass all other models, including tuned and ensembled baselines, by a significant margin, gaining 72 Elo points over our previous Real-TabPFN-2.5 tuned and ensembled. TabPFN-3-Plus (Thinking), leveraging test-time computation, significantly outperforms open-source TabPFN-3 on TabArena, beating any non-TabPFN model (including tuned and ensembled baselines) by over 200 Elo points, and outperforming AutoGluon 1.5 extreme, a complex ensemble of models including TabPFN v2, tuned for 4 hours, by over 100 Elo points while being 10x faster. Looking at the win rate matrix in Figure 12, we can see that TabPFN-3-Plus with Thinking mode (respectively TabPFN-3) has over 93% (respectively 80%) win rate against tuned

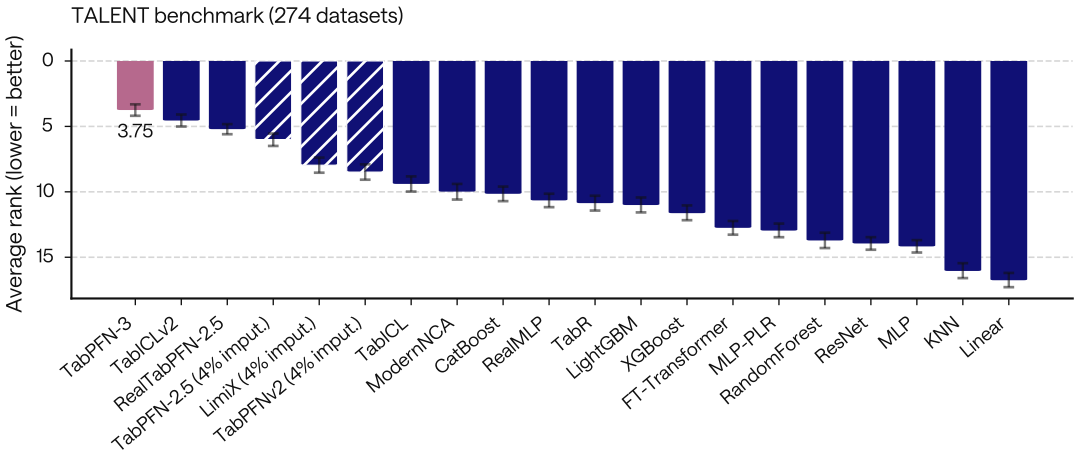



**Figure 13. Average rank on the TALENT benchmark, using the TabICLv2 evaluation protocol from Qu et al. [30] (274 datasets).** The original 300-dataset TALENT [36] minus the 26 development datasets used for TabPFN-2 / TabICLv2 development removed in the TabICLv2 paper), spanning regression, binary and multiclass classification. Bars show mean rank (lower is better); error bars are 95% bootstrap confidence intervals over datasets (see appendix E.3). Methods tagged *(N imputed, X%)* failed on some datasets and have that fraction of their score cells filled with K-nearest-neighbour values.

and ensembled CatBoost, LightGBM and XGBoost, and a 69% (respectively 56%) win rate against AutoGluon 1.5 extreme tuned for 4 hours.

**Dominating the time / performance Pareto-frontier.** The strong results of our models are achieved while being much faster to train than the baselines. On Figure 11, we can see that our model family, (TabPFN-3 with 1, 2, and 4 estimators and TabPFN-3-Plus with Thinking mode) strictly dominates the combined training + inference time/performance pareto-frontier on TabArena by a large margin.

**Scaling to larger datasets.** TabPFN-3 was built to scale to large datasets, and TabPFN-3-Plus (Thinking) benefits from this scalability. While TabArena only contains datasets up to 100k rows, we can still observe very strong performance on the 15 largest datasets in TabArena with between 10k and 100k rows, as shown in Figure 1. In particular, on this subset TabPFN-3 outperforms any other model by 100 Elo, and TabPFN-3-Plus (Thinking) dramatically outperforms any other non-TabPFN model (including tuned and ensembled baselines) by over 420 Elo points, and beats AutoGluon 1.5 extreme (4h) by 220 Elo points. Looking at the win rate matrix in Figure 3, TabPFN-3-Plus (Thinking) has over 99% win rate against tuned and ensembled LightGBM and XGBoost, 98% win rate against CatBoost tuned and ensembled, and 82% win rate against AutoGluon 1.5 extreme tuned for 4 hours. In Section 3.2.1, we study the performance of our model beyond 100K rows, going up to 1M training rows.

### 3.1.2 TALENT

The TALENT benchmark [36] provides a complementary view on the performance of TabPFN-3. Instead of a smaller curated list of datasets, this benchmark uses a large number of diverse datasets (300) from a wide range of domains. The strong results of TabPFN-3 on this benchmark confirm the robustness of its performance. Indeed, TabPFN-3 ranks first on the TALENT benchmark in aggregate, as shown in Figure 13, as well as for each task type (regression, binary and multiclass classification) in Figure 28.

### 3.1.3 TabSTAR

The TabSTAR study [37] assembled 50 text-tabular datasets, gathered from previous work [47–49]. These datasets represent real world tasks, where at least one feature is text-based and cannot be faithfully represented without text processing methods. While the open-source version of TabPFN-3 only supports numerical and categorical variables, TabPFN-3-Plus also offers native support for text features. We

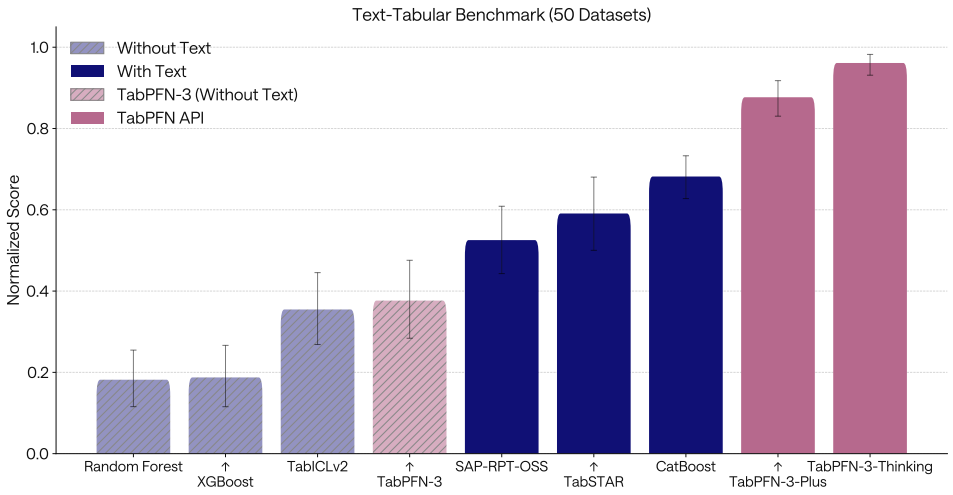



**Figure 14. Performance over the TabSTAR Text-Tabular Collection**. TabPFN-3-Plus (Thinking) and TabPFN-3-Plus significantly outperform text-aware models such as CatBoost, TabSTAR and SAP-RPT-OSS. In turn, these models dominate over numerical-only baselines, for which TabPFN-3 gets the best results.

compare TabPFN API models with both text-aware models and numerical-only baselines. Figure 14 shows that TabPFN-3-Plus dominates the leaderboard by a significant margin, and combining our thinking mode with native text support pushes performance further. Furthermore, among models that omit text features due to lack of native support, TabPFN-3 remains the top performer. Appendix E.4 provides further details on the benchmark, as well as a performance breakdown by task type.

## 3.2 Internal Benchmarks

To complement the public TabArena [1] and TALENT [36] benchmarks, we evaluate TabPFN-3 on a set of internal benchmarks designed to stress capabilities that are only partially covered by existing public evaluations. These benchmarks test whether TabPFN-3 pushes the frontier of tabular foundation models beyond the small- and medium-data regimes emphasized in prior work. In particular, we evaluate scaling to more than one million samples, high-dimensional feature spaces, many-class classification, and quantile regression.

Our primary comparisons are against the leading gradient boosted tree frameworks XGBoost [40], CatBoost [38], and LightGBM [39], as well as TabICLv2 [30], a recent foundation model for tabular data with strong results on public benchmarks.

### 3.2.1 Large Data

**Evaluation Protocol.** The primary baselines for our large-data evaluation are tree-based methods, which recent large-scale tabular benchmarks have shown to be highly competitive beyond 100,000 samples [36]. Our large-data benchmarking effort focuses on datasets with 100,000 to 1 million training rows and up to 200 features.

This benchmark targets the large-row regime for which TabPFN-3 was designed. As described in Section 2.1, TabPFN-3 first compresses feature information into fixed-dimensional row representations and subsequently performs in-context learning over these rows. This architectural decomposition enables inference on datasets with up to one million rows on a single GPU. At the same time, it induces a scaling trade-off: when both the number of rows and the number of features are very large, the early compression of feature information can become a bottleneck. We treat the high-dimensional, low-sample regime as a separate evaluation setting, studied in Section 3.2.3, rather than conflating it with the large-row setting considered here.

Our benchmark datasets span diverse real-world domains including healthcare, finance, logistics, and environmental science. For regression, the datasets in our benchmark exhibit temporal structure, where models are trained on past data and must generalize to future data. We found this setting to be the most

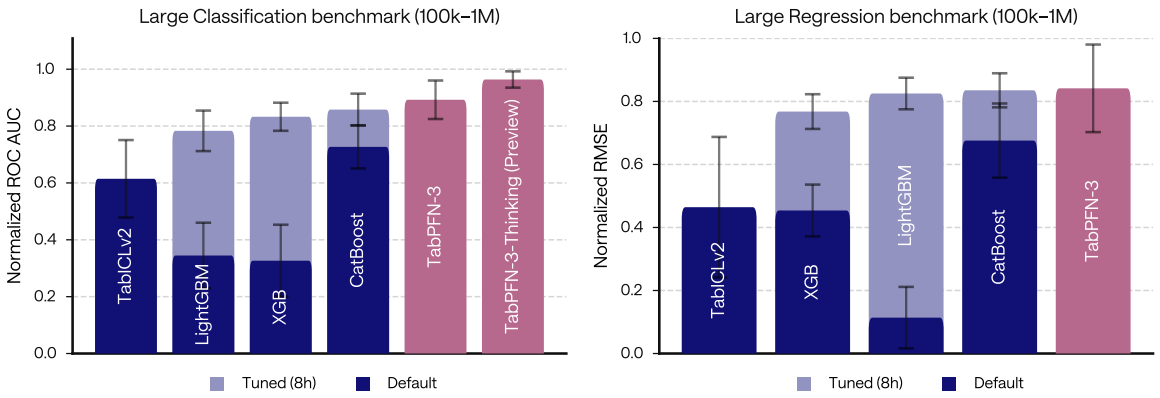



**Figure 15. TabPFN-3 achieves state-of-the-art performance on the large-rows benchmark (up to 1M training rows and 200 features, 13 datasets)**, outperforming both default and 8-hour-tuned gradient-boosted tree baselines as well as TabICLv2 in a single forward pass. **(a)** Classification (9 datasets). **(b)** Regression (4 datasets) use temporal splits. Normalized scores are higher-is-better; see Section F.1 for the normalization procedure and Appendix F.2 for critical difference diagrams.

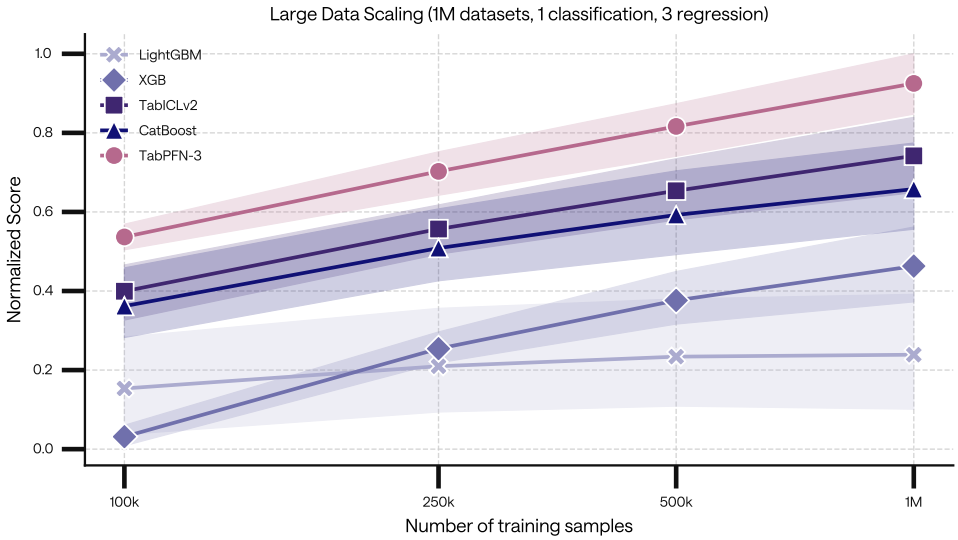



**Figure 16. TabPFN-3 tops the normalized scaling curves for ROC-AUC OvR classification and RMSE regression across dataset scales.** Results are shown on the four large-data benchmark datasets that reach at least 1M training rows (one classification, three regression). For each dataset we subsample the training set to 100k, 250k, 500k and 1M rows with 3 random repeats. Shaded bands are 95% bootstrap confidence intervals across the four datasets and 3 repeats. .

common and representative of real-world deployment conditions.

**Results.** TabPFN-3 achieves state-of-the-art performance on our large-data benchmark, outperforming default and 8-hour-tuned gradient-boosted tree baselines in a single forward pass, as shown in Figure 15. Further, we show a preview version of TabPFN-3-Plus (Thinking) on large data, which improves TabPFN-3 performance further for classification datasets (as TabPFN-3-Plus with Thinking mode does not yet support temporal datasets as of the time of writing, we could not evaluate it on our regression benchmark). To better understand how TabPFN-3 performance scales with training size, we report performance on subsampled versions of our datasets (keeping test set constant, and only considering datasets with 1M training samples) in Figure 16. Across the 100k–1M range, TabPFN-3 scales smoothly and retains the top normalized score at every training-set size.

**Large data results from TALENT benchmark.** To confirm our internal results, we also extract the 14 available datasets in the TALENT benchmark with more than 100K and less than 1M training samples

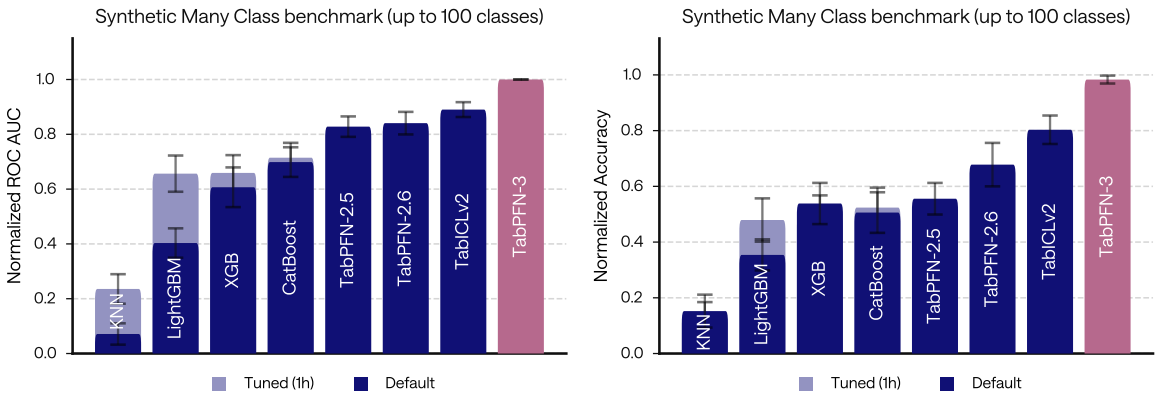



**Figure 17. On the synthetic many-class benchmark TabPFN-3 achieves a normalized ROC-AUC (OvR) of** 1.00**, outperforming all GBT baselines by a large margin**. The benchmark contains up to 100 classes, 9 datasets that are derived from TabArena regression tasks via Dirichlet-jittered quantile binning with shuffled labels). Normalized scores are higher-is-better; see Section F.1 for the normalization procedure. The corresponding Critical Difference diagram can be seen in Figure 36.

(see Appendix F.2). On this subset, TabPFN-3 is again the best ranked model against the baselines provided by the TALENT benchmark, as shown in Figure 30.

### 3.2.2 Many-Class Classification

TabPFN-3 introduces a many-class decoder (Section 2.2) that we trained to support up to 160 classes, a regime where most tabular foundation models fail entirely. Creating a benchmark from real-world datasets with naturally many classes is challenging; we therefore evaluate on a synthetic benchmark derived by bucketing regression targets from real regression benchmark datasets. We also confirm the strong performance of TabPFN-3 on the 4 datasets from the TALENT benchmark that have more than 50 classes in Section E.3.3.

**Synthetic many-class benchmark.** We construct a synthetic benchmark by converting the TabArena regression datasets into many-class classification problems via jittered quantile binning; full construction details are given in Appendix F.3. Figure 17 shows the ROC-AUC (OvR) and accuracy. TabPFN-3 achieves the highest normalized ROC-AUC of 1.00, ranking first overall and outperforming all baselines by a large margin. On ROC-AUC (OvR), the next best model is TabICLv2 at 0.89 using its many-class wrapper to go beyond its 10 classes limit. TabPFN-2.5 achieves 0.83, using its own many-class error-correcting-code-based wrapper[2]. Conventional tree-based methods and KNN all perform notably worse, even after 1 hour of tuning.

### 3.2.3 Many Features

The high-dimensional, low-sample regime poses a qualitatively different challenge from the large-row setting studied in Section 3.2.1. Whereas large-row benchmarks primarily test scalability to many training examples, the many-features setting tests robust generalization and feature-subset selection when the number of candidate features far exceeds the number of samples.

We evaluate this setting on a dedicated *many-features* slice of six real-world classification datasets with 100–320 samples, 1,100–22,200 features, and 2–4 classes, mostly from biomedical or gene-expression-style domains. Such large feature-to-sample ratios are challenging for tree-based methods because they increase the risk of selecting spurious feature interactions.

Figure 18 shows that TabPFN-3 performs strongly on this challenging slice, reaching the best normalized ROC-AUC with 32 estimators. Earlier TabPFN variants, in particular Real-TabPFN-2.5 and TabPFN v2, also perform competitively, suggesting that TabPFN-style pretraining provides a robust inductive bias for high-dimensional, low-sample problems.

[2] https://github.com/PriorLabs/tabpfn-extensions/tree/main/src/tabpfn_extensions/many_class

As described in Section 2.3, each TabPFN-3 estimator is restricted to at most 200 input features per default. Thus, for datasets with tens of thousands of raw features, individual estimators operate on feature subsets rather than compressing the full feature set. At the same estimator budget, Real-TabPFN-2.5 can slightly outperform TabPFN-3; we hypothesize that this reflects two factors: Real-TabPFN-2.5 uses up to 500 features per estimator, providing broader feature-space coverage on some datasets, and its alternating row-wise and feature-wise attention may better exploit the selected feature subset. For TabPFN-3, increasing the number of estimators improves coverage of the raw feature space and raises the probability that informative feature subsets are included. In our OSS version, this estimator budget is scaled automatically for high-dimensional inputs, making the ensemble substantially more effective in this regime.

Overall, the many-features slice suggests that TabPFN estimators can be ensembled effectively in a high-noise feature-selection regime, where conventional tree-based methods are prone to overfitting to noisy or spurious feature interactions.

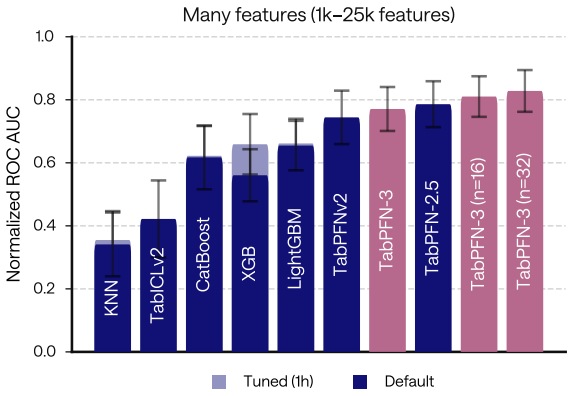



**Figure 18. TabPFN scales well to high-dimensional, low-sample classification.** Normalized ROC-AUC on the many-features benchmark slice, consisting of 6 classification datasets with 102–322 samples and 1,117–22,215 features. This high-dimensional, low-sample regime is particularly challenging for standard tree-based baselines. Increasing the number of TabPFN-3 estimators improves feature-space coverage and substantially boosts performance.


**Figure 19. TabPFN-3 exhibits strong predictive distribution modeling on quantile regression.** Normalized pinball loss on our quantile regression benchmark, constructed from TabArena regression datasets and averaged across 10 quantile levels $q \in \{0.1, 0.2, \ldots, 0.9\}$ [50]. Normalized scores are higher-is-better; see Section F.1 for the normalization procedure.

### 3.2.4 Quantile Regression

Beyond point predictions, TabPFN-3 provides full predictive distributions via a bar-distribution regression head (Section C), from which arbitrary quantiles are decoded at inference by inverting the predicted CDF — all from a single forward pass, with no retraining per quantile level. Since TabArena does not natively support quantile regression evaluation, we construct a dedicated benchmark by downloading the TabArena regression datasets and evaluating all models on pinball loss [50], averaged across 10 quantile levels $q \in \{0.1, 0.2, \ldots, 0.9\}$. We compare against four baselines spanning the typical strategies for quantile regression: a linear quantile regressor, which fits a separate pinball-loss model per quantile level; XGBoost in quantile mode, which uses a single multi-output booster but adds one tree per quantile per boosting round, scaling training cost roughly linearly in the number of levels; quantile random forests [51], which train a single MSE-objective forest and read off all quantiles from leaf-level empirical CDFs at no extra training cost; and TabICL-v2, a tabular foundation model with a quantile head.

TabPFN-3 achieves a normalized pinball loss score very close to 1.00, ranking first overall and outperforming all baselines, demonstrating that the bar-distribution head produces well-calibrated predictive distributions superior to dedicated quantile regression baselines at no additional training cost per quantile level. The normalized Pinball loss is shown in Figure 19, while the corresponding Critical Difference plot can be found in the Appendix in Figure 35.

### 3.3 Time-Series Forecasting

In addition to the classification and regression checkpoints, we release a new TabPFN-3 checkpoint for TabPFN-TS [19] fine-tuned on **synthetic** time-series data for probabilistic time-series forecasting. This checkpoint can be used in our `tabpfn-time-series` library. We evaluate it on fev-bench [52], a benchmark containing 100 diverse time-series forecasting tasks. Following this benchmark, we report win rates and skill scores relative to the Seasonal Naive baseline in Table 1 (full version in Appendix Table 17).

**Table 1.** Forecasting performance on fev-bench (100 tasks), sorted by skill score. **TabPFN-TS-3 ranks 2nd among foundation models on both SQL and MASE skill score while being trained only on synthetic data.** The full 18-baseline leaderboard can be found in Appendix H.

**(a) SQL (probabilistic)**

| Model | Win (%) | Skill (%) | Runtime (s) | Leak. (%) | # fails |
|---|---|---|---|---|---|
| Chronos-2 | 91.7 | 47.3 | 0.8 | 0 | 0 |
| TabPFN-TS-3 | 73.6 | 43.1 | 234.6 | 0 | 0 |
| TiRex | 83.4 | 42.6 | 0.2 | 1 | 0 |
| TimesFM-2.5 | 78.6 | 42.2 | 1.9 | 10 | 0 |
| Toto-1.0 | 71.6 | 40.7 | 22.1 | 8 | 0 |
| TabPFN-v2-TS | 64.1 | 39.6 | 88.9 | 0 | 2 |
| Moirai-2.0 | 66.2 | 39.3 | 0.3 | 28 | 0 |
| Chronos-Bolt | 66.2 | 38.9 | 0.2 | 0 | 0 |
| Sundial-Base | 47.1 | 33.4 | 8.0 | 1 | 0 |
| TabICL-v2 | 53.8 | 30.8 | 64.7 | 0 | 0 |
| Stat. Ensemble | 43.8 | 20.2 | 148.6 | 0 | 11 |
| Seasonal Naive | 19.1 | 0.0 | 0.5 | 0 | 0 |

**(b) MASE (point)**

| Model | Win (%) | Skill (%) | Runtime (s) | Leak. (%) | # fails |
|---|---|---|---|---|---|
| Chronos-2 | 86.9 | 35.5 | 0.8 | 0 | 0 |
| TabPFN-TS-3 | 69.8 | 30.6 | 234.6 | 0 | 0 |
| TimesFM-2.5 | 74.9 | 30.2 | 1.9 | 10 | 0 |
| TiRex | 76.9 | 30.0 | 0.2 | 1 | 0 |
| Toto-1.0 | 66.3 | 28.2 | 22.1 | 8 | 0 |
| TabPFN-v2-TS | 58.5 | 27.6 | 88.9 | 0 | 2 |
| Moirai-2.0 | 61.4 | 27.3 | 0.3 | 28 | 0 |
| Chronos-Bolt | 60.7 | 26.5 | 0.2 | 0 | 0 |
| Sundial-Base | 53.4 | 24.7 | 8.0 | 1 | 0 |
| Stat. Ensemble | 46.7 | 15.7 | 148.6 | 0 | 11 |
| TabICL-v2 | 33.2 | 7.0 | 64.7 | 0 | 0 |
| Seasonal Naive | 20.0 | 0.0 | 0.5 | 0 | 0 |

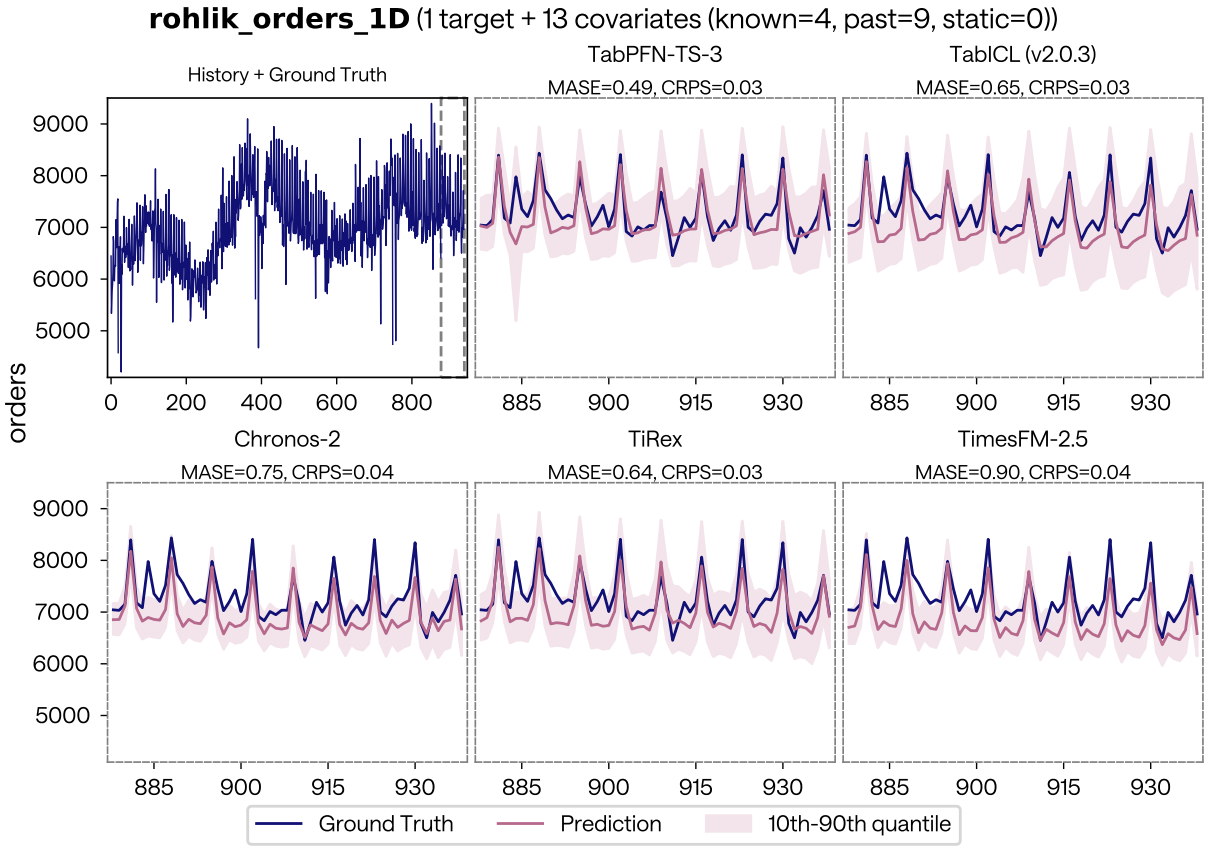



**Figure 20. Qualitative forecast comparison on a fev-bench task (`rohlik_order_1D`).** Each model column shows the forecast horizon (zoomed to time 880-935) against the held-out ground truth, with the shaded band indicating the 10th-90th quantile. The leftmost panel shows the full training history. MASE and CRPS scores are reported per model. Additional examples, including covariate panels, are in Section H.

Our checkpoint is evaluated with up to 32k historical time steps of context, well beyond the budgets typically used by patch- or window-based time-series foundation models. Compared to the original

[2]The fev-bench authors report 28.8 MASE skill for the original TabPFN-TS [52]; our re-run in Table 17 yields 27.6 — we report our own re-run for like-for-like comparison across the cohort.

TabPFN-TS [19] as evaluated by the fev-bench authors (39.6 SQL skill, 28.8 MASE skill; Shchur et al. 52), our fine-tuned variant improves to **43.1 SQL skill** and **30.6 MASE skill**. On the full 100-task cohort it ranks 2nd on mean SQL skill scores (ahead of TiRex and TimesFM-2.5) and 2nd on MASE (ahead of TimesFM-2.5, which has 10% flagged train/test leakage, and TiRex), in both cases behind only Chronos-2. Looking at the win-rate results, TabPFN-TS-3's ranking drops to the 4th place, although we found these rates to be very sensitive to tiny differences on a few datasets.

The strong performance of TabPFN-TS-3 is particularly noteworthy seeing that it is trained purely on synthetic data, while most other time-series models, including Chronos-2 [53], TiRex [54] and TimesFM-2.5 [55] are trained on real-world data. This property of our model prevents many issues from real-data pretraining: historical series are leaky and frequently recirculated across forecasting libraries (fev-bench flags 10% leakage in TimesFM-2.5 and 28% in Moirai-2.0; see Table 1), forecasting the future from historical pretraining is fundamentally out-of-distribution, and the supply of public real-world time-series data is finite, so any model relying on it inherits both its biases and its ceiling. Our synthetic prior by design has zero contamination from any specific real time series.

We also show qualitative examples in Figure 20 to give a better intuition of our model forecasts. Appendix H complements this section with the full leaderboards (Table 17), pairwise comparisons (Figure 45), additional qualitative forecasts and per-task SQL results.

## 3.4 Relational Data

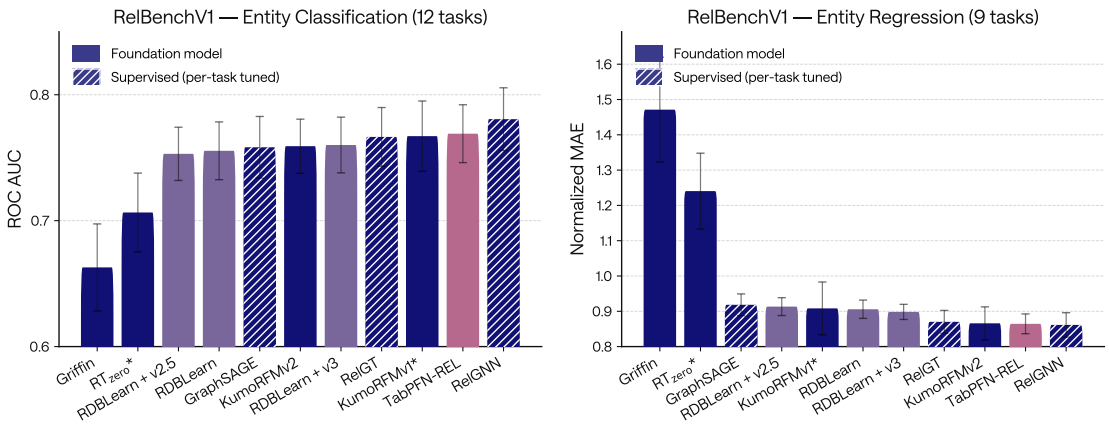



**Figure 21. TabPFN-3 tops performance on RelBenchV1 among foundation models.** Following Hudovernik et al. [56], we report the mean ROC AUC for entity classification and MAE scores for entity regression normalized by LightGBM's MAE. RelGNN [57] achieves SOTA performance on both tasks, followed by TabPFN-REL, which sets a new SOTA for foundation models. Methods marked with * in their name (KumoRFMv1, $\text{RT}_{\text{zero}}$) indicate methods that are likely following a different evaluation protocol than the one outlined in RelBench, which overestimates model performance.

Real-world data is often relational: commercial enterprises, healthcare systems, and financial institutions routinely store their core operational data across multiple interconnected tables in relational databases. Unlocking predictive insights from such data is therefore of substantial practical importance, and requires to reason jointly over heterogeneous tables linked by complex foreign-key relationships. This has motivated the development of dedicated relational foundation models (RFMs) that aim to provide accurate, up-to-date predictions via In-Context Learning (ICL) without the need for costly per-task model training and hyperparameter tuning.

This has sparked the emergence of dedicated solutions for relational data, e.g., fully supervised solutions particularly tailored for relational data such as GraphSAGE [58], RelGT [59] and RelGNN [57], closed-source relational foundation models like KumoRFMv1 [60] and KumoRFMv2 [56], as well as open-source RFMs, Griffin [61] and $\text{RT}_{\text{zero}}$ [62]. Recently, RDBLearn [63] has shown that TFMs including TabPFN can be converted into RFMs by automatically flattening the underlying database into a table.

In this section, we build on this research and show how *TabPFN-REL* using TabPFN-3 achieves state of the art performance on the popular RelBenchV1 [64] benchmark for entity classification and regression.

For RelBench, we follow the general guidelines by truncating each database at the pre-specified test timestamp before constructing the featurization and context for all test entities. Following Hudovernik et al. [56], we generally report baseline results as provided by the authors of the methods to ensure well-tuned baselines. For methods that likely follow a different evaluation regime, we rerun the evaluation using RelBench's data regime, falling back to author-reported numbers where rerunning is not possible due to model deprecation or missing checkpoints (as is the case for KumoRFMv1 and $\text{RT}_{\text{zero}}$); we note that these may not be directly comparable due to potentially different data setups. For KumoRFMv2 we adapt the original scripts provided by the authors and use four estimators and a context size of 10000 (the respective maxima for each), which we found to slightly outperform the script defaults of one estimator and a context size of 5000 samples. We compare three different versions of RDBLearn: Vanilla RDBLearn that tunes over a range of different TFMs including TabPFN-2.5, as well as versions which forgo the tuning and use either TabPFN-2.5 or TabPFN-3 as a fixed TFM.[3]

**TabPFN-REL sets a new state-of-the-art among RFMs.** We report the aggregate performance of the different RFMs and fully-supervised baselines in Figure 21 both for entity classification and entity regression on RelBenchV1, as well as per-dataset results in subsection E.5. *TabPFN-REL achieves state-of-the-art performance among RFMs on both tasks*, with KumoRFMv1/v2 coming second on regression/classification. We attribute KumoRFMv1's strong classification results in part to a potentially different evaluation regime used by the authors, which likely overestimates performance, especially on the `rel-f1` task suite. We also observe that RDBLearn with the fixed TabPFN-3 backend consistently outperforms the original RDBLearn, which itself tunes over various TFMs including TabPFN-2.5. RDBLearn using TabPFN-3 hence Pareto-dominates vanilla RDBLearn in terms of runtime and performance, and to the best of our knowledge sets a *new state-of-the-art among open-source RFMs. At the time of writing, TabPFN-3 therefore powers both the best overall relational foundation model (TabPFN-REL) and the best open-source alternative (RDBLearn + v3).*

**Comparison to fully-supervised baselines.** The fully-supervised RelGNN outperforms TabPFN-REL, with the gap being larger on classification than regression. On regression, the gap between RelGNN and TabPFN-REL is slim, with TabPFN-REL achieving lower mean rank than RelGNN. RelGT and GraphSAGE fall behind TabPFN-REL both in terms of normalized score and rank. We note that training supervised methods is several orders of magnitude more expensive than the in-context learning performed in TabPFN-REL [56, 60, 63]. This is both because training a single supervised model takes significantly longer than the forward pass of TabPFN-REL, and because supervised methods require extensive per-dataset hyperparameter tuning to achieve optimal performance. For example, we identified at least seven axes of variability in RelGNN's per-dataset configs, yielding thousands of possible hyperparameter combinations to search over.

## 3.5 Causal Inference

We follow up on our previous results [18], which showed strong performance of TabPFN-2.5 as a meta (T/X/S) learner [65] on the RealCause benchmark, by providing an evaluation on the `scikit-uplift` benchmark [66]. In terms of QINI-score, a real-world evaluation strategy for experimental data, we observe that all TabPFN-3 meta-learners improve over TabPFN-2.5, with the top two spots occupied by T and S-Learners (Figure 27). In contrast, we observe slightly worse performance compared to TabPFN-2.5 on RealCause [67]. We provide a more in-depth analysis of the results and description of the QINI evaluation protocol in Appendix E.1.

## 3.6 Embeddings

Finally, we demonstrate that TabPFN-3 generates semantically-meaningful embeddings. We follow the approach developed by Ye et al. [27] for TabPFN v2: we partition the dataset into cross-validation folds, and take the embeddings from the test-portion of the dataset in each fold. The embeddings we capture are the output of the ICL layers at the end of Stage 3 of our model (see Section /2.1 for more details).

[3]The reported results were produced with early checkpoints that did not undergo the full training pipeline and separate binary from multiclass classification. They can be identified on HuggingFace by the `20260417_<TASK_TYPE>` suffix.

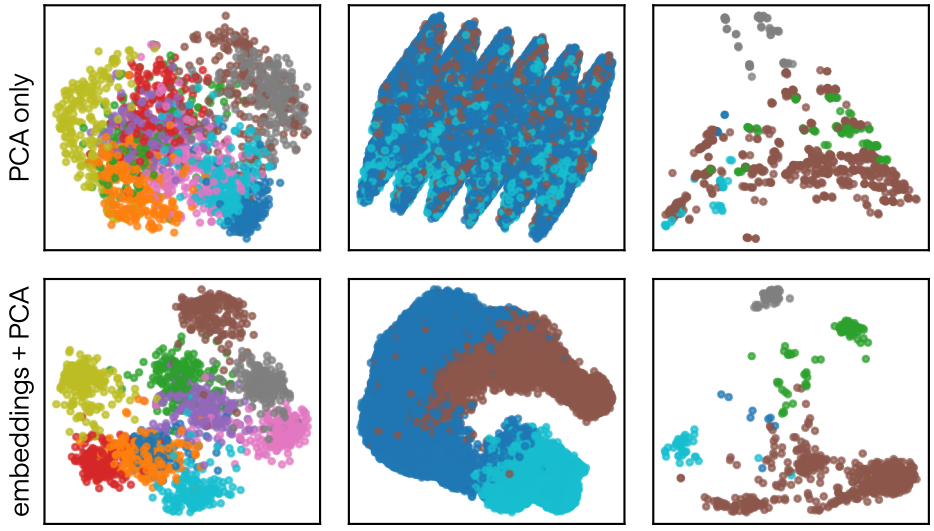



**Figure 22. TabPFN-3 extracts semantically-meaningful row embeddings.** The upper plots show 2D PCA applied directly to three classification datasets, where each point is a row, while the lower plots show PCA applied to embeddings of the rows. Color indicates the class. We observe that the embeddings are clustered by class.

Figure 22 shows that this approach continues to work well for TabPFN-3, with the generated embeddings capturing the dataset structure.

# 4 Adoption

TabPFN-3 is shipped into an already sprawling ecosystem. Since the v2 release, TabPFN has been picked up across academic ML research, applied science, and enterprise deployment. A substantial portion of the extension work referenced throughout this report (time-series, causal inference, relational data, interpretability) was driven by that community rather than initiated internally. This section describes the shape of that adoption – where the model is in production, where it is being evaluated, which platforms make it accessible, and which research areas have published applications – to give the v3 release its actual operational context.

## 4.1 Community and Open-Source Ecosystem

The open-source `tabpfn` package has surpassed 3.2 million PyPI downloads, and the original TabPFN Nature paper [17] has been cited in over 1,000 papers in the sixteen months since publication.[4] A Discord community of over 2,000 users and hundreds of resolved GitHub issues have driven cross-platform stability work, edge-case fixes, and the maturation of the model from research artifact to production-grade library.

A separate `tabpfn-extensions` repository[5] hosts community-driven extensions that compose with the core model: SHAP and SHAP-IQ interpretability, synthetic data generation and missing-value imputation, TabPFN-based feature selection, regression-via-classification, survival analysis and conditional randomization tests. TabPFN-3's reduced KV cache and inference improvements (Section 2) directly accelerate every extension that depends on repeated forward passes – most notably interpretability and conditional independence testing.

TabPFN also serves as a foundational layer for methods published as independent research, spanning time-series forecasting [19], node classification on graphs [24, 25], evolving data streams [68], causal inference [20–22], reinforcement learning [28], high-dimensional Bayesian optimization [23], and multimodal encoding [69]. As shown in Section 3, many of these extensions move further forward when run with

[4] Google Scholar entry and pepy.tech `tabpfn` download statistics, both accessed May 8, 2026.
[5] https://github.com/PriorLabs/tabpfn-extensions

TabPFN-3 as the backend rather than v2.5 or v2.6.

### 4.2 Enterprise Engagements

TabPFN has been deployed and evaluated across a wide range of enterprise settings. Examples include: *Hitachi Rail* deploys TabPFN for predictive maintenance on the Spanish rail network; in initial deployment, TabPFN reduced root-mean-square error by approximately 40% compared to their existing baseline [70]. *Creditplus Bank*, part of the Crédit Agricole group, will use distilled TabPFN models (Section 2.4.3) for assisting CPU-based credit decisioning in motor finance under appropriate credit-risk regulatory constraints [71]. *Oxford Cancer Analytics* applies TabPFN to proteomic liquid-biopsy data for early lung-disease detection [72]. A longer list of enterprise and commercial engagements is available on the Prior Labs website.

### 4.3 Platform Availability

TabPFN is available through the open-source PyPI distribution for evaluation and non-commercial use, and through a managed API for commercial workloads. The model is currently listed on the *AWS SageMaker Marketplace*[6] and the *Azure AI Foundry Model Catalog*[7], with full support for batch and real-time inference on classification and regression tasks; the TabPFN-3 release on both marketplaces follows this report. A reference integration for *Databricks* is available through the Databricks Industry Solutions repository[8]. See Section 5 for license terms, commercial-use scope, and the contact path for production deployment.

### 4.4 Research Adoption Across Domains

In addition to commercial engagement, we have collected more than 200 published research applications of TabPFN across a broad range of areas; the full list is in Appendix I.

Adoption is strongest in *healthcare and life sciences* (98 applications), reflecting TabPFN's relative advantage in data-scarce settings: diagnosis, prognosis, treatment-response prediction, biomarker modeling, survival analysis, drug discovery, pharmacokinetics, radiomics, omics, and multimodal clinical data. *Manufacturing and industrial* applications (41 papers) span concrete and asphalt strength prediction, geotechnical modeling, tunnel construction, steel and semiconductor properties, IIoT intrusion detection, rotating-machinery fault classification, battery and circuit modeling, and materials discovery. *Energy and utilities* (24 papers) cluster around environmental monitoring, renewable-energy and geophysical prediction, water and climate systems, and industrial process optimization. *Financial services* (7 papers) include transaction analytics, churn prediction, return forecasting, actuarial modeling, and credit-risk prediction; the relatively small published count almost certainly underrepresents commercial traction in a domain that publishes little. The remaining 32 applications span uncertainty estimation, hypothesis testing, Shapley value estimation, graph node classification, cybersecurity, geoscience, agriculture, soil and lunar-regolith analysis, fuel-blend prediction, crop-yield forecasting, forensic ancestry prediction, and synthetic tabular data generation.

The distribution of these applications – weighted toward domains characterized by limited, expensive, or heterogeneous data – is consistent with the regime TabPFN was designed for, and is the empirical basis for the v3 capability choices described in Section 2.

## 5 License and Availability

We release TabPFN-3 under the `TABPFN-3.0 License v1.0`, designed to be permissive for academic use, research, and evaluation in commercial settings. The license *explicitly allows* testing, evaluation, and internal benchmarking, so an organization can download the model and run preliminary assessments on its own datasets without a commercial agreement.

[6] https://aws.amazon.com/marketplace/pp/prodview-chfhncrdzlb3s
[7] https://ai.azure.com/catalog/models/TabPFN-2.5
[8] https://github.com/databricks-industry-solutions/tabpfn-databricks

The key restriction is that the model, its derivatives, and its outputs cannot be used for commercial or production purposes. This includes, but is not limited to, revenue-generating products, competitive benchmarking for procurement decisions, client deliverables, and using model outputs as inputs to internal commercial decision-making.

For production use, we offer a *Commercial Enterprise License*, available for our managed API, Virtual Private Cloud deployments (at the time of publication: AWS SageMaker & Azure AI Foundry), and on-prem or other custom deployment modes across other software platforms such as Databricks and SAP. The Commercial Enterprise License provides access to our proprietary high-speed inference engine, dedicated support, integration tooling, additional internal models, and the TabPFN-3-Plus (Thinking) variant, which is not available as part of the open-source release. The managed API runs on our optimized GPU infrastructure and is the recommended option for users without dedicated local GPUs; it is accessible via a Python SDK[9] (`pip install tabpfn-client`) or a standard REST API.

The full `TABPFN-3.0 License v1.0` text is available at https://huggingface.co/Prior-Labs/tabpfn_3/blob/main/LICENSE. For commercial licensing inquiries, please contact sales@priorlabs.ai.

[9]The Python client SDK is available on PyPI: https://github.com/PriorLabs/tabpfn-client.

# References


[1] Nick Erickson, Lennart Purucker, Andrej Tschalzev, David Holzmüller, Prateek Mutalik Desai, Frank Hutter, et al. Tabarena: A living benchmark for machine learning on tabular data. *arXiv preprint arXiv:2506.16791*, 2025.

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

[3] Katharine E Henry, David N Hager, Peter J Pronovost, and Suchi Saria. A targeted real-time early warning score (trewscore) for septic shock. *Science translational medicine*, 7(299):299ra122–299ra122, 2015.

[4] Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

[5] Yaakov Ophir, Refael Tikochinski, Christa SC Asterhan, Itay Sisso, and Roi Reichart. Deep neural networks detect suicide risk from textual facebook posts. *Scientific reports*, 10(1):16685, 2020.

[6] Hussein A Abdou and John Pointon. Credit scoring, statistical techniques and evaluation criteria: a review of the literature. *Intelligent systems in accounting, finance and management*, 18(2-3):59–88, 2011.

[7] Amir E Khandani, Adlar J Kim, and Andrew W Lo. Consumer credit-risk models via machine-learning algorithms. *Journal of Banking & Finance*, 34(11):2767–2787, 2010.

[8] Stefan Lessmann, Bart Baesens, Hsin-Vonn Seow, and Lyn C Thomas. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European journal of operational research*, 247(1):124–136, 2015.

[9] Thyago P Carvalho, Fabrízzio AAMN Soares, Roberto Vita, Roberto da P Francisco, João P Basto, and Symone GS Alcalá. A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & industrial engineering*, 137:106024, 2019.

[10] Jovani Dalzochio, Rafael Kunst, Edison Pignaton, Alecio Binotto, Srijnan Sanyal, Jose Favilla, and Jorge Barbosa. Machine learning and reasoning for predictive maintenance in industry 4.0: Current status and challenges. *Computers in industry*, 123:103298, 2020.

[11] Pierre Baldi, Peter Sadowski, and Daniel Whiteson. Searching for exotic particles in high-energy physics with deep learning. *Nature communications*, 5(1):4308, 2014.

[12] Alexander Dunn, Qi Wang, Alex Ganose, Daniel Dopp, and Anubhav Jain. Benchmarking materials property prediction methods: the matbench test set and automatminer reference algorithm. *npj Computational Materials*, 6(1):138, 2020.

[13] Ravid Shwartz-Ziv and Amitai Armon. Tabular data: Deep learning is not all you need. *Information fusion*, 81:84–90, 2022.

[14] Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in neural information processing systems*, 35: 507–520, 2022.

[15] David Salinas and Nick Erickson. Tabrepo: A large scale repository of tabular model evaluations and its automl applications. In *AutoML Conference 2024 (ABCD Track)*, 2024.

[16] Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*, 2022.

[17] Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025. ISSN 1476-4687. doi: 10.1038/s41586-024-08328-6. URL https://doi.org/10.1038/s41586-024-08328-6.

[18] Léo Grinsztajn, Klemens Flöge, Oscar Key, Brendan Roof Felix Birkel, Phil Jund, Benjamin Jäger, Adrian Hayler, Dominik Safaric, Felix Jablonski Simone Alessi, Mihir Manium, Rosen Yu, Anurag Garg, Jake Robertson, Shi Bin (Liam) Hoo, Vladyslav Moroshan, Magnus Bühler, Lennart Purucker, Clara Cornu, Lilly Charlotte Wehrhahn, Alessandro Bonetto, Sauraj Gambhir, Noah Hollmann, and Frank Hutter. Tabpfn-2.5: Advancing the state of the art in tabular foundation models, 2025.

[19] Shi Bin Hoo, Samuel Müller, David Salinas, and Frank Hutter. The tabular foundation model tabpfn outperforms specialized time series forecasting models based on simple features. In *NeurIPS Workshop on Time Series in the Age of Large Models*, 2024.

[20] Jake Robertson, Arik Reuter, Siyuan Guo, Noah Hollmann, Frank Hutter, and Bernhard Schölkopf. Do-pfn: In-context learning for causal effect estimation. *arXiv preprint arXiv:2506.06039*, 2025.

[21] Vahid Balazadeh, Hamidreza Kamkari, Valentin Thomas, Benson Li, Junwei Ma, Jesse C. Cresswell, and Rahul G. Krishnan. Causalpfn: Amortized causal effect estimation via in-context learning, 2025. URL https://arxiv.org/abs/2506.07918.

[22] Yuchen Ma, Dennis Frauen, Emil Javurek, and Stefan Feuerriegel. Foundation models for causal inference via prior-data fitted networks, 2025. URL https://arxiv.org/abs/2506.10914.

[23] Rosen Ting-Ying Yu, Cyril Picard, and Faez Ahmed. Git-bo: High-dimensional bayesian optimization with tabular foundation models. *arXiv preprint arXiv:2505.20685*, 2025. doi: 10.48550/arXiv.2505.20685. URL https://arxiv.org/abs/2505.20685.

[24] Adrian Hayler, Xingyue Huang, İsmail İlkan Ceylan, Michael Bronstein, and Ben Finkelshtein. Bringing graphs to the table: Zero-shot node classification via tabular foundation models. *arXiv preprint arXiv:2509.07143*, 2025. doi: 10.48550/arXiv.2509.07143. URL https://arxiv.org/abs/2509.07143.

[25] Dmitry Eremeev, Gleb Bazhenov, Oleg Platonov, Artem Babenko, and Liudmila Prokhorenkova. Turning tabular foundation models into graph foundation models, 2025. URL https://arxiv.org/abs/2508.20906.

[26] David Rundel, Julius Kobialka, Constantin von Crailsheim, Matthias Feurer, Thomas Nagler, and David Rügamer. Interpretable machine learning for tabpfn. In *World Conference on Explainable Artificial Intelligence*, pages 465–476. Springer, 2024.

[27] Han-Jia Ye, Si-Yang Liu, and Wei-Lun Harry Chao. A closer look at tabpfn v2: Understanding its strengths and extending its capabilities. *Advances in Neural Information Processing Systems*, 38: 135605–135637, 2026.

[28] David Schiff, Ofir Lindenbaum, and Yonathan Efroni. Gradient free deep reinforcement learning with tabpfn. *arXiv preprint arXiv:2509.11259*, 2025. doi: 10.48550/arXiv.2509.11259. URL https://arxiv.org/abs/2509.11259.

[29] Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A tabular foundation model for in-context learning on large data. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=0VvD1PmNzM.

[30] Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICLv2: A better, faster, scalable, and open tabular foundation model. In *International Conference on Machine Learning*, 2026.

[31] Ken M. Nakanishi. Scalable-softmax is superior for attention, 2025. URL https://arxiv.org/abs/2501.19399.

[32] David Holzmüller, Léo Grinsztajn, and Ingo Steinwart. Better by default: Strong pre-tuned mlps and boosted trees on tabular data. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/2ee1c87245956e3eaa71aaba5f5753eb-Abstract-Conference.html.

[33] Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. FlashAttention-3: fast and accurate attention with asynchrony and low-precision. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NeurIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

[34] Maximilian Muschalik, Hubert Baniecki, Fabian Fumagalli, Patrick Kolpaczki, Barbara Hammer, and Eyke Hüllermeier. shapiq: Shapley interactions for machine learning. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=knxGmi6SJi.

[35] Philip Boeken and Joris M. Mooij. Dynamic structural causal models, 2024. URL https://arxiv.org/abs/2406.01161. UAI 2024 Workshop on Causal Inference for Time Series Data.

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

[37] Alan Arazi, Eilam Shapira, and Roi Reichart. TabSTAR: A Tabular Foundation Model for Tabular Data with Text Fields. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen, editors, *Advances in Neural Information Processing Systems*, volume 38, pages 172108–172161. Curran Associates, Inc., 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/faf6e23e198314c7728eaa6ac44ae079-Paper-Conference.pdf.

[38] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31, 2018.

[39] 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. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 3146–3154. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf.

[40] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

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

[42] Han-Jia Ye, Huai-Hong Yin, De-Chuan Zhan, and Wei-Lun Chao. Revisiting nearest neighbor for tabular data: A deep tabular baseline two decades later. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=JytL2MrlLT.

[43] Daniel Beaglehole, David Holzmüller, Adityanarayanan Radhakrishnan, and Mikhail Belkin. xrfm: Accurate, scalable, and interpretable feature learning models for tabular data, 2025. URL https://arxiv.org/abs/2508.10053.

[44] Junwei Ma, Valentin Thomas, Rasa Hosseinzadeh, Hamidreza Kamkari, Alex Labach, Jesse C. Cresswell, Keyvan Golestan, Guangwei Yu, Anthony L. Caterini, and Maksims Volkovs. Tabdpt: Scaling tabular foundation models on real data, 2025. URL https://arxiv.org/abs/2410.18164.

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

[46] Xiyuan Zhang, Danielle C. Maddix, Junming Yin, Nick Erickson, Abdul Fatir Ansari, Boran Han, Shuai Zhang, Leman Akoglu, Christos Faloutsos, Michael W. Mahoney, Cuixiong Hu, Huzefa Rangwala, George Karypis, and Bernie Wang. Mitra: Mixed synthetic priors for enhancing tabular foundation models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=t8YRsWY6HM.

[47] Xingjian Shi, Jonas Mueller, Nick Erickson, Mu Li, and Alexander J Smola. Benchmarking multimodal automl for tabular data with text fields. *arXiv preprint arXiv:2111.02705*, 2021.

[48] Léo Grinsztajn, Edouard Oyallon, Myung Jun Kim, and Gaël Varoquaux. Vectorizing string entries for data processing on tables: when are larger language models better? *arXiv preprint arXiv:2312.09634*, 2023.

[49] Myung Jun Kim, Léo Grinsztajn, and Gaël Varoquaux. Carte: pretraining and transfer for tabular learning. *arXiv preprint arXiv:2402.16785*, 2024.

[50] Roger Koenker and Gilbert Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978. ISSN 00129682, 14680262. URL http://www.jstor.org/stable/1913643.

[51] Nicolai Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7:983–999, 2006.

[52] Oleksandr Shchur, Abdul Fatir Ansari, Caner Turkmen, Lorenzo Stella, Nick Erickson, Pablo Guerron, Michael Bohlke-Schneider, and Yuyang Wang. fev-bench: A realistic benchmark for time series forecasting. *arXiv preprint arXiv:2509.26468*, 2025.

[53] Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, Mononito Goswami, Shubham Kapoor, Danielle C. Maddix, Pablo Guerron, Tony Hu, Junming Yin, Nick Erickson, Prateek Mutalik Desai, Hao Wang, Huzefa Rangwala, George Karypis, Yuyang Wang, and Michael Bohlke-Schneider. Chronos-2: From univariate to universal forecasting, 2025. URL https://arxiv.org/abs/2510.15821.

[54] Andreas Auer, Patrick Podest, Daniel Klotz, Sebastian Böck, Günter Klambauer, and Sepp Hochreiter. TiRex: Zero-Shot Forecasting Across Long and Short Horizons with Enhanced In-Context Learning. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2505.23719.

[55] Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 10148–10167. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/das24c.html.

[56] Valter Hudovernik, Federico López, Vid Kocijan, Akihiro Nitta, Jan Eric Lenssen, Jure Leskovec, and Matthias Fey. Kumorfm-2: Scaling foundation models for relational learning, 2026. URL https://arxiv.org/abs/2604.12596.

[57] Tianlang Chen, Charilaos Kanatsoulis, and Jure Leskovec. Relgnn: Composite message passing for relational deep learning, 2025. URL https://arxiv.org/abs/2502.06784.

[58] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs, 2018. URL https://arxiv.org/abs/1706.02216.

[59] Vijay Prakash Dwivedi, Sri Jaladi, Yangyi Shen, Federico López, Charilaos I. Kanatsoulis, Rishi Puri, Matthias Fey, and Jure Leskovec. Relational graph transformer, 2026. URL https://arxiv.org/abs/2505.10960.

[60] Matthias Fey, Vid Kocijan, Federico Lopez, Jan Eric Lenssen, and Jure Leskovec. Kumorfm: A foundation model for in-context learning on relational data. *Kumo.ai*, 2025. URL https://kumo.ai/research/kumo_relational_foundation_model.pdf.

[61] Yanbo Wang, Xiyuan Wang, Quan Gan, Minjie Wang, Qibin Yang, David Wipf, and Muhan Zhang. Griffin: Towards a graph-centric relational database foundation model, 2025. URL https://arxiv.org/abs/2505.05568.

[62] Rishabh Ranjan, Valter Hudovernik, Mark Znidar, Charilaos Kanatsoulis, Roshan Upendra, Mahmoud Mohammadi, Joe Meyer, Tom Palczewski, Carlos Guestrin, and Jure Leskovec. Relational transformer: Toward zero-shot foundation models for relational data, 2026. URL https://arxiv.org/abs/2510.06377.

[63] Yanlin Zhang, Linjie Xu, Quan Gan, David Wipf, and Minjie Wang. Rdblearn: Simple in-context prediction over relational databases, 2026. URL https://arxiv.org/abs/2602.18495.

[64] Joshua Robinson, Rishabh Ranjan, Weihua Hu, Kexin Huang, Jiaqi Han, Alejandro Dobles, Matthias Fey, Jan E. Lenssen, Yiwen Yuan, Zecheng Zhang, Xinwei He, and Jure Leskovec. Relbench: A benchmark for deep learning on relational databases, 2024. URL https://arxiv.org/abs/2407.20060.

[65] Sören R. Künzel, Jasjeet S. Sekhon, Peter J. Bickel, and Bin Yu. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the National Academy of Sciences*, 116(10):4156–4165, 2019.

[66] Irina Elisova Maksim Shevchenko. User guide for uplift modeling and casual inference. https://www.uplift-modeling.com/en/latest/user_guide/index.html, 2020.

[67] Brady Neal, Chin-Wei Huang, and Sunand Raghupathi. Realcause: Realistic causal inference benchmarking. *CoRR*, abs/2011.15007, 2020. URL https://arxiv.org/abs/2011.15007.

[68] Afonso Lourenço, João Gama, Eric P. Xing, and Goreti Marreiros. In-context learning of evolving data streams with tabular foundational models. *arXiv preprint arXiv:2502.16840*, 2025. doi: 10.48550/arXiv.2502.16840. URL https://arxiv.org/abs/2502.16840.

[69] Jiaqi Luo, Yuan Yuan, and Shixin Xu. Time: Tabpfn-integrated multimodal engine for robust tabular-image learning, 2025. URL https://arxiv.org/abs/2506.00813.

[70] Prior Labs. Predictive Maintenance for Rail Networks: Hitachi Rail Case Study. https://priorlabs.ai/case-studies/hitachi, 2026. Accessed May 2026.

[71] Prior Labs. Credit Decisioning at Creditplus Bank: Case Study. https://priorlabs.ai/case-studies/credit-plus, 2026. Accessed May 2026.

[72] Prior Labs. Clinical Decision Support with Oxford Cancer Analytics: Case Study. https://priorlabs.ai/case-studies/oxcan, 2026. Accessed May 2026.

[73] Paul W Holland. Statistics and causal inference. *Journal of the American Statistical Association*, 81(396):945–960, 1986.

[74] Arpad E Elo. The proposed uscf rating system, its development, theory, and applications. *Chess life*, 22(8):242–247, 1967.

[75] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez, et al. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*, 2024.

[76] Maksim A. Terpilowski. scikit-posthocs: Pairwise multiple comparison tests in python. *Journal of Open Source Software*, 4(36):1169, 2019. doi: 10.21105/joss.01169. URL https://doi.org/10.21105/joss.01169.

[77] Dan-Ni Wu, Joey Jen, Erickson Fajiculay, Min-Fen Hsu, Ming-Chu Chang, Jen-Chen Yeh, Karen Sargsyan, Juozas Kupcinskas, Jurgita Skieceviciene, Ruta Steponaitiene, Egidijus Morkunas, Greta Gedgaudiene, Chao-Ping Hsu, Yu-Ting Chang, and Chun-Mei Hu. Panmetai - a high performance tabular foundation model for accurate pancreatic cancer diagnosis via nmr metabolomics. *Nature Communications*, 17, 2026. doi: 10.1038/s41467-026-69426-9. URL https://doi.org/10.1038/s41467-026-69426-9.

[78] Asif Adil and Stephanie Hurwitz. Deep learning models enable healthy donor management through prediction of mobilization success. *Transplantation and Cellular Therapy*, 32:S3, 2026. doi: 10.1016/j.jtct.2026.02.016. URL https://doi.org/10.1016/j.jtct.2026.02.016.

[79] Hongxin Zheng, Wenxin Gan, Yizi Liu, Shuyu Duan, Kun Li, Gongping Li, Yanqiu Xue, and Yu Xie. Differentiation between psychotic and non-psychotic major depression by the tabular prior-data fitted network. *Journal of Affective Disorders*, 403:121454, 2026. doi: 10.1016/j.jad.2026.121454. URL https://doi.org/10.1016/j.jad.2026.121454.

[80] Mohammadreza Noori Sichani, Omid Mazahery Dehkordi, Morteza Khorshidi, Amirehsan Teimortashlu, and Pourya Nejatipour. Machine learning based optimization of fly ash content for improving geopolymer concrete compressive strength. *Scientific Reports*, 15, 2025. doi: 10.1038/s41598-025-29088-x. URL https://doi.org/10.1038/s41598-025-29088-x.

[81] Huixia Zhang, Jiajun Tong, Minmin Chen, and Xichuan Cao. Boosting pre-trained model with silica nanoparticles cellular toxicity prediction. *Scientific Reports*, 16, 2026. doi: 10.1038/s41598-025-33872-0. URL https://doi.org/10.1038/s41598-025-33872-0.

[82] Yuanyuan Chen, Tianbiao Yang, Tianlu Chen, Huanming Xiao, Kejun Zhou, Keke Ding, Quan Liu, Mengci Li, Xufei Peng, Tao Sun, Xiaoning Wang, Ping Liu, Xin Deng, Zhenhua Zhang, Ka Zhang, Xianzhang Huang, Xiaoling Chi, Alice Pik-Shan Kong, Vincent Wai-Sun Wong, Wei Jia, and Guoxiang Xie. A metabolite-augmented fib-4 machine learning panel achieves superior liver fibrosis staging in chronic liver disease. *Cell Reports Medicine*, 7(4):102726, 2026. ISSN 2666-3791. doi: https://doi.org/10.1016/j.xcrm.2026.102726. URL https://www.sciencedirect.com/science/article/pii/S2666379126001436.

[83] Junaid Latif, Na Chen, Jia Xie, Zheng Ni, Lang Zhu, Azka Saleem, Kai Li, and Hanzhong Jia. Deep learning-aided prediction and mechanistic analysis of reaction kinetics in biochar-catalyzed antibiotic degradation. *Biochar*, 8, 2026. doi: 10.1007/s42773-026-00606-y. URL https://doi.org/10.1007/s42773-026-00606-y.

[84] Fatemeh Shoaei, Mohammad Pishdar, Mozafar Bag-Mohammadi, and Mojtaba Karami. Lroo rug pull detector: A leakage-resistant framework based on on-chain and osint signals, 2026. URL https://arxiv.org/abs/2603.11324.

[85] Yicheng Wang and Sandro Claudio Lera. Meta-learning for return prediction in shifting market regimes. *Journal of Financial Markets*, page 101042, 2025. doi: 10.1016/j.finmar.2025.101042. URL https://doi.org/10.1016/j.finmar.2025.101042.

[86] Luis Magadán, José Roldán-Gómez, Juan Carlos Granda, and Francisco José Suárez. Early fault classification in rotating machinery with limited data using TabPFN. *IEEE Sensors Journal*, 23 (24):30960–30970, 2023. doi: 10.1109/JSEN.2023.3331100. URL https://ieeexplore.ieee.org/document/10318062.

[87] Karim K. Ben Hicham, Jan G. Rittig, Martin Grohe, and Alexander Mitsos. Tabular foundation models for in-context prediction of molecular properties, 2026. URL https://arxiv.org/abs/2604.16123.

[88] Samuel J Tingle, Georgios Kourounis, Sofia Kazerouni, Harry VM Spiers, Miguel Larraz, Maithili Mehta, Serena MacMillan, Sarah A Hosgood, Michael L Nicholson, Neil S Sheerin, and Colin H Wilson. Combining bulkformer and tabpfn to predict post- transplant function from kidney biopsies during machine perfusion or cold storage. Preprint at Research Square, 2026. URL https://doi.org/10.21203/rs.3.rs-9242336/v1.

[89] Dmitrii Seletkov, Paul Hager, Rickmer Braren, Daniel Rueckert, and Raphael Rehms. Survival in-context: Prior-fitted in-context learning tabular foundation model for survival analysis, 2026. URL https://arxiv.org/abs/2603.29475.

[90] Kazi Sakib Hasan and Irfan Sadi Dhrubo. Advancing cardiovascular disease diagnosis with an interpretable and responsible ai framework. *Scientific Reports*, 2026. doi: 10.1038/s41598-026-35451-3. URL https://doi.org/10.1038/s41598-026-35451-3.

[91] Kosuke Kita, Yuki Suzuki, Takashi Fujimoto, Keisuke Uemura, Yoshito Otake, Masayuki Furuya, Yuya Kanie, Tomohiro Wataya, Daiki Nishigaki, Junya Sato, Miyuki Tomiyama, Noriyuki Tomiyama, Seiji Okada, Masatoshi Hori, and Takahito Fujimori. Transformer-based multimodal model for estimation of appendicular lean mass using incomplete chest radiographs and electronic health record. *Journal of Translational Medicine*, 24, 2026. doi: 10.1186/s12967-026-08079-0. URL https://doi.org/10.1186/s12967-026-08079-0.

[92] J. Villines, R. Stirnimann, L. P. Lukas, O. Taran, M. Tuci, Y. Li, C. R. Jutzeler, J.L.K. Kramer, F. H. Geisler, D. Bourbeau, R. James Cotton, and S. C. Brüningk. The asia data science challenge: Predicting functional and neurological recovery from acute isncsci scores. *Topics in Spinal Cord Injury Rehabilitation*, pages 1–12, 2026. doi: 10.46292/sci25-00137. URL https://doi.org/10.46292/sci25-00137.

[93] Mona Schoberth, Samuel Böhm, Oleg Borisov, Yong Li, Gabriele Greve, Bayram Edemir, Owen M. Woodward, Hyun Jun Jung, Frank Hutter, Lukas Westermann, Anna Köttgen, Pascal Schlosser, Michael Köttgen, and Stefan Haug. Transcriptome-based cell type assignment for kidney cell culture models. *bioRxiv*, 2026. doi: 10.64898/2026.03.30.715265. URL https://www.biorxiv.org/content/early/2026/04/01/2026.03.30.715265.

[94] Miglionico Pasquale, Matic Marin, Franchini Luca, Hiroki Arai, Nemati Fard Lorenzo Amir, Arora Chakit, Magda Gherghinescu, Natalia De Oliveira Rosa, Kise Ryoji, J. Silvio Gutkind, Cesare Orlandi, Asuka Inoue, and Raimondi Francesco. Computed atlas of the human gpcr-g protein signaling complexes. *bioRxiv*, 2026. doi: 10.64898/2026.03.07.710286. URL https://www.biorxiv.org/content/early/2026/03/10/2026.03.07.710286.

[95] F. Schwarz, L. Levien, M. Maulhardt, G. Wulf, N. Brökers, and E. Aydilek. Predicting adverse events for risk stratification of chemotherapy based stem cell mobilization in multiple myeloma. *npj Digital Medicine*, 9, 2026. doi: 10.1038/s41746-026-02394-y. URL https://doi.org/10.1038/s41746-026-02394-y.

[96] Dianne Daniels, Kfir Cohen, David Last, Shirley Sharabi, Maayan Zuniga, Nora Lahat, Renata Faermann, Osnat Halshtok, Anat Shalmon, David Samoocha, Michael Gotlieb, Yael Mardor, and Miri Sklair-Levy. Application of treatment response assessment maps (trams), based on delayed-contrast mri for radiomic characterization of breast lesions. *Scientific Reports*, 16, 2026. doi: 10.1038/s41598-026-40472-z. URL https://doi.org/10.1038/s41598-026-40472-z.

[97] GitHub - Smallriver2024/STBNet: TabPFN-Based Interpretable Deep Learning Model for Discriminating Spinal Tuberculosis from Pyogenic Spinal Infection — github.com. https://github.com/{S}mallriver2024/{S}{T}{B}{N}et, . [Accessed 11-05-2026].

[98] Ane G Domingo-Aldama, Marcos Merino Prado, Alain García Olea, Josu Goikoetxea, Koldo Gojenola, and Aitziber Atutxa. Automating early disease prediction via structured and unstructured clinical data, 2026. URL https://arxiv.org/abs/2603.28167.

[99] Chunlai Fang, Ning Ma, and Limin Qian. Multi-task transformer framework and radiomic signatures for multi-lesion segmentation, detection, and grading in diabetic retinopathy. *Photodiagnosis and Photodynamic Therapy*, page 105455, 2026. doi: 10.1016/j.pdpdt.2026.105455. URL https://doi.org/10.1016/j.pdpdt.2026.105455.

[100] Jinling Liu, Xudi Pang, Huiming Cao, Yuzhen Sun, and Yong Liang. Mucb-tabpfn: A multimodal feature fusion framework for predicting human blood concentrations of organic pollutants. *Ecotoxicology and Environmental Safety*, 314:120055, 2026. doi: 10.1016/j.ecoenv.2026.120055. URL https://doi.org/10.1016/j.ecoenv.2026.120055.

[101] Hao Zhong, Ping Xiong, Nannan Wang, Kunda Li, Ruifeng Wang, Yiyang Wu, and Defang Ouyang. Physics-based machine learning for enhanced drug formulation development. *Journal of Controlled Release*, 394:114860, 2026. doi: 10.1016/j.jconrel.2026.114860. URL https://doi.org/10.1016/j.jconrel.2026.114860.

[102] GitHub - Rishabhmannu/MultiModal-Stress-Detection-ML: Advanced machine learning pipeline for real-time stress detection using synchronized chest and wrist wearable sensors, featuring state-of-the-art TabPFN models and interpretable cross-modal attention mechanisms with clinical-grade reporting. — github.com. https://github.com/{R}ishabhmannu/{M}ulti{M}odal-{S}tress-{D}etection-{M}{L}, . [Accessed 11-05-2026].

[103] Woruo Chen, Yao Tian, Nian Liao, Youchao Deng, Dejun Jiang, and Dongsheng Cao. Tabpfn opens new avenues for small-data tabular learning in drug discovery. *Journal of Chemical Information and Modeling*, 66:3525–3539, 2026. doi: 10.1021/acs.jcim.5c02823. URL https://doi.org/10.1021/acs.jcim.5c02823.

[104] Minh-Khoi Pham, Thang-Long Nguyen Ho, Thao Thi Phuong Dao, Tai Tan Mai, Minh-Triet Tran, Marie Elizabeth Ward, Una Geary, Rob Brennan, Nick McDonald, Martin Crane, and Marija Bezbradica. Retrieval-aligned tabular foundation models enable robust clinical risk prediction in electronic health records under real-world constraints. Preprint at Research Square, 2026. URL https://doi.org/10.21203/rs.3.rs-9085469/v1.

[105] Ibrahim Sadek, Shafiq Ul Rehman, Ahmed Gehad, Esraa G. Eltasawi, Ahmed AbdelKader, Rawan Abdelnasser, Dina Nashaat, Raef Mourad Zaki, and Lamees N. Mahmoud. From raw clinical data to robust prediction: an ai framework for early lymphedema detection. *BMC Medical Research Methodology*, 26, 2026. doi: 10.1186/s12874-026-02805-4. URL https://doi.org/10.1186/s12874-026-02805-4.

[106] Eric Johnsson, Shrinjay Sharma, Arvind Gangoli Rao, David Dubbeldam, Sofia Calero, and Thijs J. H. Vlugt. Predicting the maximum loading in zeolites for hydroisomerization applications: A machine learning approach. *The Journal of Physical Chemistry C*, 130:4299–4314, 2026. doi: 10.1021/acs.jpcc.5c08611. URL https://doi.org/10.1021/acs.jpcc.5c08611.

[107] Daqi Chen, Anping Liu, Xiaoxia Wang, Xiaoming Liu, Wenjie Liang, Linsheng Luo, Hua Nie, and Xingming Zhong. A multidimensional clinical prediction model for early screening of recurrent spontaneous abortion: integrating coagulation, immune, and endocrine markers. *Frontiers in Immunology*, 17, 2026. doi: 10.3389/fimmu.2026.1774359. URL https://doi.org/10.3389/fimmu.2026.1774359.

[108] Zhe Huang, Weishen Pan, Shudhanshu Alishetti, Ashley N. Beecy, Zhenzhen Liu, Albert Gong, Saebyeol Shin, Kevin J. Clerkin, Rochelle L. Goldsmith, David T. Majure, Chris Kelsey, David vanMaanan, Jeffrey Ruhl, Naomi Tesfuzigta, Erica Lancet, Deepa Kumaraiah, Gabriel Sayer, Deborah Estrin, Kilian Weinberger, Nir Uriel, and Fei Wang. Multimodal multi-instance learning for cardiopulmonary exercise testing performance prediction. *npj Digital Medicine*, 9, 2026. doi: 10.1038/s41746-026-02493-w. URL https://doi.org/10.1038/s41746-026-02493-w.

[109] Ruobing Liu, Mohamed Azzam, Nikki Zabik, Shibiao Wan, Jennifer Blackford, and Jieqiong Wang. Classification of adolescent drinking via behavioral, biological, and environmental features: A machine learning approach with bias control. Preprint at medRxiv, 2026. URL https://doi.org/10.64898/2026.02.24.26347002.

[110] Jin Mu, Zheng-Zheng Tang, and Guanhua Chen. Systematic benchmarking of foundation models and classical baselines for microbiome-based disease prediction. Preprint at Research Square, 2026. URL https://doi.org/10.21203/rs.3.rs-8912605/v1.

[111] Dimitrios Papakyriakopoulos, Pantelis Z. Lappas, and Manolis N. Kritikos. Heart: Hierarchical ensemble model using augmented representations and tabular learning for coronary artery disease prediction. Preprint at Research Square, 2026. URL https://doi.org/10.21203/rs.3.rs-8239358/v1.

[112] Russell Dinnage and Dan Warren. A niche in the machine: The promise of ai foundation models for species distribution modeling. Preprint at EcoEvoRxiv, 2026. URL https://doi.org/10.32942/x2vq10.

[113] Wall Kim, Chaeyoung Song, and Hanul Kim. Multimodalpfn: Extending prior-data fitted networks for multimodal tabular learning, 2026. URL https://arxiv.org/abs/2602.20223.

[114] Michael Backenköhler, Joschka Groß, and Andrea Volkamer. Chempfn: Unified bayesian modelling of bioactivities across chembl. Preprint at ChemRxiv, 2026. URL https://doi.org/10.26434/chemrxiv.15000292/v1.

[115] GitHub - SindyPin/TACO: TabPFN Augmented Causal Outcomes for Early Detection of Long COVID — github.com. https://github.com/{S}indy{P}in/{T}{A}{C}{O}, . [Accessed 11-05-2026].

[116] GitHub - AhmedAlMarouf/FoundationModel_on_Mimic3_ClinRisk: This project in on Clinical risk prediction on MIMIC-III using **TabPFN** (Prior-Fitted Networks). — github.com. https://github.com/{A}hmed{A}l{M}arouf/{F}oundation{M}odel_on_{M}imic3_{C}lin{R}isk, . [Accessed 11-05-2026].

[117] Hang Yu, Sina Saffaran, Roberto Tonelli, John G. Laffey, Qingchen Zhang, Antonio M. Esquinas, Lucas Martins de Lima, Letícia Kawano-Dourado, Israel S. Maia, Alexandre Biasi Cavalcanti, Enrico Clini, and Declan G. Bates. Evaluating the effect of heart and respiratory rate measurement errors on the ability to predict the outcome of high flow nasal cannula therapy: a multi-centre study. *Critical Care*, 29, 2025. doi: 10.1186/s13054-025-05765-1. URL https://doi.org/10.1186/s13054-025-05765-1.

[118] Lei Miao, He Zhao, Xiaowu Zhang, Jingui Li, Qing Peng, Yingen Luo, Pengfei Tian, Xuefeng Luo, Jun Tie, and Xiao Li. Enhancing post-tips hepatic encephalopathy risk stratification: a hybrid tabpfn model leveraging radiomics, deep transfer learning features, and meld score. *Hepatology International*, 20:428–440, 2025. doi: 10.1007/s12072-025-10934-z. URL https://doi.org/10.1007/s12072-025-10934-z.

[119] Xiaoyu Wu, Rui Zheng, Quan Liu, and Jianwen Jiang. Digital discovery of synthesizable metal-organic frameworks via molecular dynamics-informed, high-fidelity deep learning. *Advanced Functional Materials*, 36, 2025. doi: 10.1002/adfm.202519565. URL https://doi.org/10.1002/adfm.202519565.

[120] Theresa Maurer, Lennart Purucker, Frank Hutter, Peter Pfaffelhuber, and Carola Sophia Heinzel. Enhancing intra-continental biogeographical ancestry prediction through a machine learning marker selection method. Preprint at bioRxiv, 2025. URL https://doi.org/10.1101/2025.11.08.687358.

[121] Tianzhu Liu, Huanjun Wang, Yan Guo, Yongsong Ye, Bei Weng, Xiaodan Li, Jun Chen, Shanghuang Xie, Guimian Zhong, Zhixuan Song, and Lesheng Huang. Tabular prior-data fitted network in real-world ct radiomics: benign vs. malignant renal tumor classification. *Quantitative Imaging in Medicine and Surgery*, 15:10847–10861, 2025. doi: 10.21037/qims-2025-1132. URL https://doi.org/10.21037/qims-2025-1132.

[122] Jifan Gao, Michael Rosenthal, Brian Wolpin, and Simona Cristea. Count-based approaches remain strong: A benchmark against transformer and llm pipelines on structured ehr, 2025. URL https://arxiv.org/abs/2511.00782.

[123] Md Rakibul Hasan, Md Zakir Hossain, Aneesh Krishna, Shafin Rahman, and Tom Gedeon. Privacy-preserving empathy detection in video interactions, 2025. URL https://arxiv.org/abs/2504.10808.

[124] Jerome Sepin. Multiple imputation of a continuous outcome with fully observed predictors using tabpfn. *Stats*, 9:38, 2026. doi: 10.3390/stats9020038. URL https://doi.org/10.3390/stats9020038.

[125] Wenlei Fan, Yuejin Zhang, Kejian Fu, Zheng Shen, Xinran Li, and Xiong Li. Lightweight and interpretable integrated diagnostic framework for skin lesion segmentation and classification. *Information Sciences*, 745:123429, 2026. doi: 10.1016/j.ins.2026.123429. URL https://doi.org/10.1016/j.ins.2026.123429.

[126] George Obaido and Ebenezer Esenogho. Evaluating eeg-based seizure classification using foundation and classical ensemble models. *Applied Sciences*, 16:3120, 2026. doi: 10.3390/app16073120. URL https://doi.org/10.3390/app16073120.

[127] Xuming Kang, Yanfang Zhao, Zhijun Tan, Lin Yao, and Yingying Guo. Enhancing kelp origin prediction accuracy: A tabpfn model with stable isotope analysis and explainability techniques for robust insights. *Food Chemistry*, 509:148591, 2026. doi: 10.1016/j.foodchem.2026.148591. URL https://doi.org/10.1016/j.foodchem.2026.148591.

[128] Mohamed Riad Youcefi, Saad Alatefi, Menad Nait Amar, and Ahmad Alkouh. Accurate prediction of co2 frosting temperature in natural gas mixtures using explainable data-driven frameworks. *Chemometrics and Intelligent Laboratory Systems*, 272:105679, 2026. doi: 10.1016/j.chemolab.2026.105679. URL https://doi.org/10.1016/j.chemolab.2026.105679.

[129] Vladimir Chupakhin and John DiBella. Descriptor-first approach for admet prediction in the polarishub antiviral challenge. *Journal of Chemical Information and Modeling*, 66:406–412, 2025. doi: 10.1021/acs.jcim.5c02094. URL https://doi.org/10.1021/acs.jcim.5c02094.

[130] Pablo M. Granitto, Emanuela Betta, Iuliia Khomenko, Michele Pedrotti, Andrea Romano, and Franco Biasioli. On the use of tabpfn on mass spectrometry analysis of volatile organic compounds. *Scientific Reports*, 16, 2025. doi: 10.1038/s41598-025-29128-6. URL https://doi.org/10.1038/s41598-025-29128-6.

[131] Prior Labs. How bostongene utilized tabpfn to identify immune system profiles associated with immunotherapy response in cancer patients. https://www.linkedin.com/pulse/how-bostongene-utilized-tabpfn-identify-immune-system-profiles-vexle/, 2025. Online case study on TabPFN in immune profiling. Accessed 7 Nov 2025.

[132] Ryunosuke Noda, Daisuke Ichikawa, and Yugo Shibagaki. Machine learning-based diagnostic prediction of minimal change disease: Model development study. *Scientific Reports*, 14:23460, 2024. doi: 10.1038/s41598-024-73898-4. URL https://www.nature.com/articles/s41598-024-73898-4.

[133] Daniiar Dyikanov, Aleksandr Zaitsev, Tatiana Vasileva, Iris Wang, Arseniy A. Sokolov, Evgenii S. Bolshakov, and et al. Comprehensive peripheral blood immunoprofiling reveals five immunotypes with immunotherapy response characteristics in patients with cancer. *Cancer Cell*, 42(5):759–779.e12, 2024. doi: 10.1016/j.ccell.2024.04.008. URL https://www.cell.com/cancer-cell/fulltext/S1535-6108(24)00132-6.

[134] Saud A. Alzakari, Abdullah Aldrees, Muhammad Fahad Umer, Luca Cascone, Nader Innab, and Imran Ashraf. Artificial intelligence-driven predictive framework for early detection of still birth. *SLAS Technology*, 29(6):100203, 2024. doi: 10.1016/j.slast.2024.100203. URL https://www.sciencedirect.com/science/article/pii/S2472630324000852.

[135] Mert Karabacak, Alexander Schupper, Matthew Carr, and Konstantinos Margetis. A machine learning-based approach for individualized prediction of short-term outcomes after anterior cervical corpectomy. *Asian Spine Journal*, 18(4):541–549, 2024. doi: 10.31616/asj.2024.0048. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC11366553/.

[136] Vinh Quang Tran and Haewon Byeon. Predicting dementia in parkinson's disease on a small tabular dataset using hybrid lightgbm–tabpfn and shap. *Digital Health*, 10:20552076241272585, 2024. doi: 10.1177/20552076241272585. URL https://journals.sagepub.com/doi/10.1177/20552076241272585.

[137] Mert Karabacak, Burak Berksu Ozkara, Tobias D. Faizy, Trevor Hardigan, Jeremy J. Heit, Dheeraj A. Lakhani, Konstantinos Margetis, Kambiz Nael, Max Wintermark, and V. Sreenivasan Yedavalli. Data-driven prognostication in distal medium vessel occlusions using explainable machine learning. *American Journal of Neuroradiology*, 46(4):725–732, 2025. doi: 10.3174/ajnr.A8547. URL https://www.ajnr.org/content/46/4/725.

[138] Hang Yu, Sina Saffaran, Israel S. Maia, Enrico Clini, Declan G. Bates, and NIVPredict study group. Early prediction of non-invasive ventilation outcome using the tabpfn machine learning model: A multi-centre validation study. *Intensive Care Medicine*, 51(8):1542–1544, 2025. doi: 10.1007/s00134-025-08025-6. URL https://link.springer.com/article/10.1007/s00134-025-08025-6.

[139] Gahao Chen and Ziwei Yang. Clinical prediction of intravenous immunoglobulin-resistant kawasaki disease based on interpretable transformer model. *PLOS ONE*, 20(7):e0327564, 2025. doi: 10.1371/journal.pone.0327564. URL https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0327564.

[140] Moumen El-Melegy, Ahmed Mamdouh, Samia Ali, Mohamed Badawy, Mohamed A. El-Ghar, Norah S. Alghamdi, and Ayman El-Baz. Prostate cancer diagnosis via visual representation of tabular data and deep transfer learning. *Bioengineering*, 11(7):635, 2024. doi: 10.3390/bioengineering11070635. URL https://www.mdpi.com/2306-5354/11/7/635.

[141] Yunhua Li et al. Mri delta-radiomics and morphological feature-driven tabpfn model for preoperative prediction of lymphovascular invasion in invasive breast cancer. *Technology in Cancer Research & Treatment*, 24:15330338251362050, 2025. doi: 10.1177/15330338251362050. URL https://journals.sagepub.com/doi/10.1177/15330338251362050.

[142] Peng Wang, Hongjun Liu, Yiming Shi, Ao Liu, Qingyu Zhu, Irina Albu, Maja Pacholec, Lulu Cheng, Xu Sun, and Xinli Chi. Harnessing small-data machine learning for transformative mental health forecasting: Towards precision psychiatry with personalised digital phenotyping. *Med Research*, 2025. doi: 10.1002/mdr2.70017. URL https://onlinelibrary.wiley.com/doi/10.1002/mdr2.70017.

[143] Bruno-LSo. Ml-health-tabpfn. https://github.com/Bruno-LSo/ML-Health-TABPFN. GitHub repository for cardiovascular risk stratification using TabPFN. Accessed 7 Nov 2025.

[144] Yan Xu, Zheng Xu, Chenyu Li, Lingyu Xu, Xinyuan Wang, Chen Guan, Siqi Jiang, Ningxin Zhang, Minghao Gu, and Yanlu Xin. Tabular prior data fitted network predicts acute kidney injury with routine clinical data. SSRN preprint, 2025. URL https://ssrn.com/abstract=5397006.

[145] Thomas Derya Kocar, Simone Brefka, Christoph Leinert, Utz Lovis Rieger, Hans Kestler, Dhayana Dallmeier, Jochen Klenk, and Michael Denkinger. Deep learning predicts postoperative mobility, activities of daily living, and discharge destination in older adults from sensor data. *Sensors*, 25 (16):5021, 2025. doi: 10.3390/s25165021. URL https://www.mdpi.com/1424-8220/25/16/5021.

[146] Rawan AlSaad, Majid Alabdulla, Aliya Tabassum, and Rajat Thomas. From mother to infant: predicting infant temperament using maternal mental health measures and tabular machine learning models. *Frontiers in Public Health*, 13:1659987, 2025. doi: 10.3389/fpubh.2025.1659987. URL https://www.frontiersin.org/articles/10.3389/fpubh.2025.1659987.

[147] Hao Liu et al. Characterizing clinical risk profiles of major complications in type 2 diabetes mellitus using deep learning algorithms. *Frontiers in Endocrinology*, 16:1657366, 2025. doi: 10.3389/fendo.2025.1657366. URL https://www.frontiersin.org/articles/10.3389/fendo.2025.1657366.

[148] Yilang Ding, Jiawen Ren, Jiaying Lu, Gloria Hyunjung Kwak, Armin Iraji, and Alex Fedorov. Longitudinal progression prediction of alzheimer's disease with tabular foundation model. *arXiv preprint arXiv:2508.17649*, 2025. URL https://arxiv.org/abs/2508.17649.

[149] Madhushan Ramalingam. Uncertainty-aware tabular prediction: Evaluating vbll-enhanced tabpfn in safety-critical medical data. *arXiv preprint arXiv:2509.10048*, 2025. URL https://arxiv.org/abs/2509.10048.

[150] Ellen L. Larson et al. Machine learning models of rna expression landscapes help predict overall tumor response to chemotherapy in cholangiocarcinoma. *Clinical Cancer Research*, 31(13_Suppl): A020, 2025. URL https://aacrjournals.org/clincancerres/article/31/13_Supplement/A020/763312.

[151] Junwei Ma, Apoorv Dankar, George Stein, Guangwei Yu, and Anthony L. Caterini. Tabpfgen – tabular data generation with tabpfn. *arXiv preprint arXiv:2406.05216*, 2024. doi: 10.48550/arXiv.2406.05216. URL https://arxiv.org/abs/2406.05216.

[152] Sirin Cetin, Ayse Ulgen, Ozge Pasin, Hakan Sıvgın, and Meryem Cetin. Determination of malignancy risk factors using gallstone data and comparing machine learning methods to predict malignancy. *Journal of Clinical Medicine*, 14(17):6091, 2025. doi: 10.3390/jcm14176091. URL https://www.mdpi.com/2077-0383/14/17/6091.

[153] Maicon Herverton Lino Ferreira da Silva Barros et al. Machine learning classification of favorable vs unfavorable tuberculosis treatment outcomes using clinical and sociodemographic data from brazil's sinan-tb (2001–2023). Research Square preprint, 2025. URL https://www.researchsquare.com/article/rs-7502054/v1.

[154] Vinh Nguyen Dao et al. Early prediction of gestational diabetes using integrated cell-free dna features and omics-derived genetic scores. medRxiv preprint, 2025. URL https://www.medrxiv.org/content/10.1101/2025.09.03.25334985v1.

[155] Chaochao Pan et al. Sense-of-agency as clinically accessible features for schizophrenia prediction: Interpretable ensemble machine learning research and webserver development. *Asian Journal of Psychiatry*, 111:104674, 2025. doi: 10.1016/j.ajp.2025.104674. URL https://www.sciencedirect.com/science/article/pii/S187620182500317X.

[156] Jinying Zhu, Ping Xiong, Wei Wang, Tianshu Lu, and Defang Ouyang. Integrating artificial intelligence and physiologically based pharmacokinetic modeling to predict in vitro and in vivo fate of amorphous solid dispersions. *Journal of Controlled Release*, 386:114123, 2025. doi: 10.1016/j.jconrel.2025.114123. URL https://doi.org/10.1016/j.jconrel.2025.114123.

[157] Okan Düzyel, Mehmet Kuntalp, Fevzi Yasin Karabulut, and Damla Kuntalp. Tabpfn achieves superior performance in respiratory disease classification based on respiratory sound data. SSRN preprint, 2025. URL https://ssrn.com/abstract=5529540.

[158] Woruo Chen, Yao Tian, Youchao Deng, Dejun Jiang, and Dongsheng Cao. Tabpfn opens new avenues for small-data tabular learning in drug discovery. ChemRxiv preprint, 2025. URL https://chemrxiv.org/engage/chemrxiv/article-details/68d29b1cf2aff1677025b18f.

[159] Shidian Zhu, Hui Zhang, Yanlin Liu, Wenyu Bu, Qiang Wu, Jin Wang, Wandi Chen, Qiannong Wu, Zhirong Geng, and Fuming Liu. Development of an optimized risk evaluation system for cardiovascular-kidney-metabolic syndrome-associated coronary heart disease based on tabular prior-data fitted network. *Digital Health*, 11:20552076251379379, 2025. doi: 10.1177/20552076251379379. URL https://doi.org/10.1177/20552076251379379.

[160] Asif Adil et al. Advanced deep learning enables prediction of allogeneic stem cell mobilization success. bioRxiv preprint, 2025. URL https://www.biorxiv.org/content/10.1101/2025.09.17.676674v1.

[161] Mayra Pacheco-Cardín, Juan Luis Hernández-Arellano, José-Manuel Mejía-Muñoz, and Aide Aracely Maldonado-Macías. Comparison of machine learning and deep learning models in manual strength prediction using anthropometric variables. *International Journal of Occupational Safety and Ergonomics*, pages 1–10, 2025. doi: 10.1080/10803548.2025.2554461. Online ahead of print.

[162] Jie Li, Andrew McCarthy, Zhizhuo Zhang, and Stephen Young. Uncertainty-guided model selection for tabular foundation models in biomolecule efficacy prediction. *arXiv preprint arXiv:2510.02476*, 2025. URL https://arxiv.org/abs/2510.02476.

[163] R. Zheng. A multitask deep learning framework for clinical decision-making in assisted reproductive technology. Master's thesis, Massachusetts Institute of Technology, 2025. URL https://dspace.mit.edu/handle/1721.1/162969. M.Eng. thesis.

[164] Sindy Licette Piñero, Xiaomei Li, Lin Liu, Jiuyong Li, Sang Hong Lee, Marnie Winter, Thin Nguyen, Junpeng Zhang, and Thuc Duy Le. Taco: Tabpfn augmented causal outcomes for early detection of long covid. *medRxiv*, 2025. doi: 10.1101/2025.10.02.25337138. URL https://www.medrxiv.org/content/10.1101/2025.10.02.25337138v1.

[165] Tuyen Vu, Ha Xuan Tran, Lin Liu, Jiuyong Li, Jia Tina Du, and Thuc Duy Le. Foundation model-based recommendation of optimal neoadjuvant therapy in breast cancer. *medRxiv*, 2025. doi: 10.1101/2025.10.03.25337255. URL https://www.medrxiv.org/content/10.1101/2025.10.03.25337255v1.

[166] John Adeoye and Yu-Xiong Su. Artificial intelligence for predicting post-excision recurrence and malignant progression in oral potentially malignant disorders: a retrospective cohort study. *International Journal of Surgery*, 2025. doi: 10.1097/JS9.0000000000003592. Online ahead of print.

[167] H. Xu, X. Xu, K. Zhang, J. Lin, M. B. Saad, G. Eapen, J. Zhang, D. L. Gibbons, J. Heymach, A. A. Vaporciyan, J. Roth, R. Mehran, P. Balter, J. M. Pollard, D. C. Qian, S. H. Lin, S. Gandhi, Z. Liao, J. Wu, and J. Y. Chang. Vision-language ai model for detecting pet/ct-occult lymph node metastasis in early-stage nsclc treated with sabr to prevent regional recurrence. *International Journal of Radiation Oncology, Biology, Physics*, 123(1):S201, 2025. URL https://www.redjournal.org/article/S0360-3016(25)05890-0/fulltext. ASTRO Annual Meeting Abstract.

[168] Asmaa A. Mahdi. Diagnosing patient stroke status using modern ai after dataset balancing: A comprehensive comparative study. *Journal of Scientific Reports*, 9(1):219–228, 2025. doi: 10.58970/JSR.1105. URL https://www.ijsab.com/jsr-volume-9-issue-1/8205.

[169] Mathias Kirk Lausen, Sarah Steiner Clausen, Maja Holm Bak, Inger Vestergaard Kristensen, Morten Hasselstrøm Jensen, Peter Vestergaard, Sisse Heiden Laursen, and Simon Lebech Cichosz. Development of machine learning models to predict hypoglycemia and hyperglycemia on days of hemodialysis in patients with diabetes based on continuous glucose monitoring. *medRxiv*, 2025. doi: 10.1101/2025.10.24.25338707. URL https://www.medrxiv.org/content/10.1101/2025.10.24.25338707v1.

[170] Wen Wen, Tingrui Zhang, Haina Zhao, Jingyan Liu, Heng Jiang, Yushuang He, and Zekun Jiang. Multimodal model enhances qualitative diagnosis of hypervascular thyroid nodules: integrating radiomics and deep learning features based on b-mode and pdi images. *Gland Surgery*, 14(8): 1558–1571, 2025. doi: 10.21037/gs-2025-183. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC12432950/.

[171] Konstantinos Vrettos, Konstantina Kasioumi, Nikolaos Galanakis, Elias Kehagias, Nikolaos Kontopodis, Nikolas Matthaiou, and Michail E. Klontzas. Radiomics enhance the prediction of endovascular treatment success for femoropopliteal chronic total occlusions: a proof-of-concept study. *European Journal of Radiology*, 194:112496, 2025. doi: 10.1016/j.ejrad.2025.112496. URL https://pubmed.ncbi.nlm.nih.gov/41166916/.

[172] Vincent Michel Borderie, Cristina Georgeon, Nassim Louissi, Benjamin Memmi, Malika Hamrani, Nacim Bouheraoua, and Anatole Chessel. Corvis$^{ST}$ biomechanical indices in the diagnosis of corneal stromal and endothelial disorders: an artificial intelligence-based comparative study. *British Journal of Ophthalmology*, 2025. doi: 10.1136/bjo-2025-327855. URL https://pubmed.ncbi.nlm.nih.gov/41130662/. Online ahead of print.

[173] Seungeon Choi, Joshep Shin, Yunu Kim, Jaemyung Shin, and Minsam Ko. Estimating sleep-stage distribution from respiratory sounds via deep audio segmentation. *Sensors*, 25(20):6282, 2025. doi: 10.3390/s25206282. URL https://www.mdpi.com/1424-8220/25/20/6282.

[174] Xiaohui Lin, Yujia Wang, Lingling Zhang, and et al. Construction of machine learning classification prediction model for vancomycin blood concentrations based on mimic-iv database. *China Pharmacy (ZHONGGUO YAOFANG)*, 36(19):2448–2453, 2025. doi: 10.6039/j.issn.1001-0408.2025.19.16. URL https://journal.china-pharmacy.com/en/article/doi/10.6039/j.issn.1001-0408.2025.19.16/.

[175] Cinthia Fonseca Araujo, Felipe Mendes Delpino, Lílian Munhoz Figueiredo, Alexandre Dias Porto Chiavegatto Filho, Bruno Pereira Nunes, Helena Silveira Schuch, and Flavio Fernando Demarco. Predicting negative self-rated oral health in adults using machine learning: A longitudinal study in southern brazil. *Journal of Dentistry*, 163:106164, 2025. doi: 10.1016/j.jdent.2025.106164. URL https://pubmed.ncbi.nlm.nih.gov/41075925/.

[176] Christopher Kolberg, Katharina Eggensperger, and Nico Pfeifer. Tabpfn-wide: Continued pretraining for extreme feature counts. *arXiv preprint arXiv:2510.06162*, 2025. doi: 10.48550/arXiv.2510.06162. URL https://arxiv.org/abs/2510.06162.

[177] Gahao Chen and Ziwei Yang. Risk prediction for gastrointestinal bleeding in pediatric henoch–schönlein purpura using an interpretable transformer model. *Frontiers in Physiology*, 16: 1630807, 2025. doi: 10.3389/fphys.2025.1630807. URL https://www.frontiersin.org/journals/physiology/articles/10.3389/fphys.2025.1630807.

[178] Artem Sakhno, Daniil Tomilov, Yuliana Shakhvalieva, Inessa Fedorova, Daria Ruzanova, Omar Zoloev, Andrey Savchenko, and Maksim Makarenko. Financial transaction retrieval and contextual evidence for knowledge-grounded reasoning, 2026. URL https://arxiv.org/abs/2603.15459.

[179] Xia Li, Hanghang Zheng, Xiwei Zhuang, Zhong Wang, Xiao Chen, Hong Liu, Jasmine Bai, and Mao Mao. Class-imbalanced-aware adaptive dataset distillation for scalable pretrained model on credit scoring, 2025. URL https://arxiv.org/abs/2501.10677.

[180] zx20030501. Github - zx20030501/sp500-market-prediction-tabpfn: Multi-factor financial time series prediction with tabpfn. https://github.com/zx20030501/sp500-market-prediction-tabpfn, 2026. [Accessed 12-May-2026].

[181] Nicolas Leyh. Can automl handle the constraints of finance? a domain-specific benchmark of automated ml frameworks and tabpfn. In *ACIS 2025 Proceedings*. Association for Information Systems, 2025. URL https://aisel.aisnet.org/acis2025/28.

[182] Jasmin Ze Kee Chu, Joel Chia Ming Than, and Hudyjaya Siswoyo Jo. Deep learning for cross-selling health insurance classification. In *Proceedings of the 2024 International Conference on Green Energy, Computing and Sustainable Technology (GECOST)*, Miri, Sarawak, Malaysia, 2024. IEEE. URL https://ieeexplore.ieee.org/abstract/document/10475046.

[183] Zenghui Wang, Zhihong Man, Lin Meng, Shijian Cang, and Yanxia Sun. Ai-driven digital twin and delay-aware surrogate mpc framework for biogas production. *Computers & Chemical Engineering*, 210:109637, 2026. doi: 10.1016/j.compchemeng.2026.109637. URL https://doi.org/10.1016/j.compchemeng.2026.109637.

[184] Zhuowen Meng, Xin Liu, Shuang Huang, and Weiyong Zhan. How to achieve artificial aging approaching natural aging: Long-term remediation effects of biochar on cadmium contamination in soils. *Geoderma*, 468:117792, 2026. doi: 10.1016/j.geoderma.2026.117792. URL https://doi.org/10.1016/j.geoderma.2026.117792.

[185] Nan Qiao, Shuning Wang, Sijing Duan, Wenpeng Cui, Yuzhe Chen, Qingchen Yang, Xingyuan Hua, and Ju Ren. Cloud-edge collaborative large models for robust photovoltaic power forecasting, 2026. URL https://arxiv.org/abs/2603.22343.

[186] Weimin Wang, Hejuan Liu, Xilin Shi, Yunhe Su, Haizeng Pan, Shengnan Ban, and Hongwei Wang. Predicting fault gouge shear strength under small-sample and missing feature conditions: A three-stage framework with pretrained tabular inference. *Rock Mechanics and Rock Engineering*, 2026. doi: 10.1007/s00603-026-05420-3. URL https://doi.org/10.1007/s00603-026-05420-3.

[187] Yuting Yang, Gang Mei, Zhengjing Ma, Nengxiong Xu, and Jianbing Peng. Simple and robust forecasting of spatiotemporally correlated small earth data with a tabular foundation model, 2025. URL https://arxiv.org/abs/2510.08920.

[188] Tong Sun. Leveraging the tabpfn algorithm for high-resolution mapping of groundwater bicarbonate and its scaling risk across china. https://doi.org/10.6084/m9.figshare.31646935.v1, 2026. [Accessed 11-05-2026].

[189] Xun Li and Yujing Jiang. A tabpfn-based framework for slope stability analysis using geometric features and shear strength parameters. *Rock Mechanics Bulletin*, page 100326, 2026. doi: 10.1016/j.rockmb.2026.100326. URL https://doi.org/10.1016/j.rockmb.2026.100326.

[190] Bai Liu, Yun Chen, and Dazhi Yang. Evaluating tabpfn for regression tasks in solar energy meteorology. *Solar Energy*, 309:114472, 2026. doi: 10.1016/j.solener.2026.114472. URL https://doi.org/10.1016/j.solener.2026.114472.

[191] Wei Si, Zhixiong Chen, Chi Yung Jim, Mou Leong Tan, Dong Liu, Yue Yao, Lifei Wei, Shangshang Xu, and Fei Zhang. Resolving inherent constraints in eutrophication monitoring of small lakes using multi-source satellites and machine learning. *npj Clean Water*, 8, 2025. doi: 10.1038/s41545-025-00525-8. URL https://doi.org/10.1038/s41545-025-00525-8.

[192] Gang Chen, Zihan Yang, Peng Sun, Chenglong Wang, Jinliang Li, Guang Yang, and Likun Pan. Data-augmented machine learning for predicting biomass-derived hard carbon anode performance in sodium-ion batteries, 2025. URL https://arxiv.org/abs/2510.12833.

[193] Hyunseok Yang and Jungsu Park. Comparing the performance of a deep learning model (tabpfn) for predicting river algal blooms with varying data composition. *Journal of the Korean Wetlands Society*, 26(3):197–203, 2024. URL https://www.earticle.net/Article/A456244.

[194] Sadegh Khanmohammadi, Miguel G. Cruz, Daniel D. B. Perrakis, Martin E. Alexander, and Mehrdad Arashpour. Using automl and generative ai to predict the type of wildfire propagation in canadian conifer forests. *Ecological Informatics*, 82:102711, 2024. doi: 10.1016/j.ecoinf.2024.102711. URL https://www.sciencedirect.com/science/article/pii/S157495412400253X.

[195] Baha'a Zaher Saleh et al. Machine learning framework for energy consumption optimization using the tabpfnregressor algorithm. Preprint / technical report on wastewater treatment plant energy optimization, 2025. URL https://www.researchgate.net/publication/390516459_Machine_learning_framework_for_energy_consumption_optimization_using_the_TabPFNRegressor_algorithm. Details via ResearchGate preprint 390516459; please update with final publication metadata if available.

[196] Aarxshi. Rainfall_tabpfn: Post-processing rainfall forecasts with tabpfn. https://github.com/aarxshi/rainfall_tabpfn, 2024. Code repository for rainfall forecast post-processing with TabPFN.

[197] Open Climate Fix. Adjuster this! tabpfn for solar forecast error adjustment. https://gist.github.com/anshulg954/5f4423ee6b3d3151fa8d0d7fcd98d3eb, 2025. Prototype from Open Climate Fix Summer of Code project for TabPFN-based solar forecast error adjustment.

[198] Bowen Chen, Zhuo Xiong, Yongchun Zhao, and Junying Zhang. Multi-view machine learning model of ash chemical composition–minerals: Improving ash fusibility prediction and interpretability of high-alkali coal. SSRN preprint 5406504, 2025. URL https://ssrn.com/abstract=5406504.

[199] Sandeep Sharma et al. Machine learning-based predictions of henry coefficients for long-chain alkanes in one-dimensional zeolites: Application to hydroisomerization. *The Journal of Physical Chemistry C*, 2025. doi: 10.1021/acs.jpcc.5c03868. URL https://pubs.acs.org/doi/10.1021/acs.jpcc.5c03868. In press / early access; uses ML including TabPFN-style approaches for Henry coefficient prediction.

[200] Sandeep Sharma. Data and models for shape-selective adsorption in zeolites for long-chain alkane hydroisomerization. https://doi.org/10.4233/uuid:f36da034-5cb3-42ca-a53d-d351f68a9ffa, 2025. Repository associated with shape-selectivity modeling in zeolites; includes TabPFN-based components.

[201] Hao Chen et al. Coupling eur prediction with fracturing optimization: An integrated machine learning framework for shale gas development. Preprint / article as indexed via ScienceDirect (S2666519025001128), 2025. URL https://www.researchgate.net/publication/395761327_Coupling_EUR_Prediction_with_Fracturing_Optimization_An_Integrated_Machine_Learning_Framework_for_Shale_Gas_Development. Uses ML, including TabPFN-based models, for EUR prediction and fracturing design; update with final journal info when confirmed.

[202] Authors not clearly specified. Enhancing reservoir parameter prediction workflows via advanced core data augmentation. ResearchGate preprint 395434405, 2025. URL https://www.researchgate.net/publication/395434405_Enhancing_Reservoir_Parameter_Prediction_Workflows_via_Advanced_Core_Data_Augmentation. Machine learning workflow including TabPFN for improved reservoir parameter prediction; please update with definitive metadata if published.

[203] Hongyu Wang et al. Application of tabpfn model on the energy performance improvement of high-power multistage centrifugal pump. *Energy*, 2025. URL https://www.sciencedirect.com/science/article/abs/pii/S0360544225040411. Uses TabPFN-based modelling for entropy generation and efficiency optimization; see article S0360544225040411.

[204] Authors not clearly specified in the available metadata. The first 0.2 degrees resolution global continental heat flow map: Advancing fine-scale geothermal modeling. Preprint / technical report as indexed via ResearchGate, 2025. URL https://www.researchgate.net/publication/396728153_The_First_02_Resolution_Global_Continental_Heat_Flow_Map_Advancing_Fine-Scale_Geothermal_Modeling. Combines GeoClimaProx and TabPFN-style models for global heat flow estimation; please update with full author list and venue from the official publication if available.

[205] Justus Viga, Penelope Mueck, Alexander Löser, and Torben Weis. Fuelcast: Benchmarking tabular and temporal models for ship fuel consumption. *arXiv preprint arXiv:2510.08217*, 2025. doi: 10.48550/arXiv.2510.08217. URL https://arxiv.org/abs/2510.08217.

[206] Davit Aslanyan. Automated supervised identification of thunderstorm ground enhancements (tges). *arXiv preprint arXiv:2510.25125*, 2025. doi: 10.48550/arXiv.2510.25125. URL https://arxiv.org/abs/2510.25125.

[207] Arsalan Mahmoodzadeh, Manish Kewalramani, Abdulaziz Alghamdi, Anwar Ahmed, Shtwai Alsubai, Abdullah Alqahtani, Abed Alanazi, and Sivaprakasam Palani. Machine learning-based prediction of crack mouth opening displacement in ultra-high-performance concrete. *Scientific Reports*, 15, 2025. doi: 10.1038/s41598-025-23610-x. URL https://doi.org/10.1038/s41598-025-23610-x.

[208] Zhongchang Wang, Zhihao Hu, Yi Yang, and Xiaohang Tang. High-fidelity numerical assessment of overburden fracturing: a pfc2d-tabpfn-shap workflow for accurate, interpretable wcfz height prediction. *Engineering Research Express*, 8:075105, 2026. doi: 10.1088/2631-8695/ae586d. URL https://doi.org/10.1088/2631-8695/ae586d.

[209] Dimitrios Sinodinos, Bahareh Nikpour, Jack Yi Wei, Sushant Sinha, Xiaoping Ma, Kashif Rehman, Stephen Yue, and Narges Armanfard. Multitask-informed prior for in-context learning on tabular data: Application to steel property prediction, 2026. URL https://arxiv.org/abs/2603.22738.

[210] Jeffrey Hu, Rongzhi Dong, Ying Feng, Ming Hu, and Jianjun Hu. Foundation-model surrogates enable data-efficient active learning for materials discovery, 2026. URL https://arxiv.org/abs/2603.12567.

[211] Shriyank Somvanshi, Pavan Hebli, Gaurab Chhetri, and Subasish Das. Tabular data with class imbalance: Predicting electric vehicle crash severity with pretrained transformers (tabpfn) and mamba-based models. In *2025 International Conference on Machine Learning and Applications (ICMLA)*, page 1460–1465. IEEE, December 2025. doi: 10.1109/icmla66185.2025.00222. URL http://dx.doi.org/10.1109/ICMLA66185.2025.00222.

[212] Shriyank Somvanshi, Anannya Ghosh Tusti, Mahmuda Sultana Mimi, Md Monzurul Islam, Sazzad Bin Bashar Polock, Anandi Dutta, and Subasish Das. Applying mambaattention, tabpfn, and tabtransformers to classify sae automation levels in crashes, 2025. URL https://arxiv.org/abs/2506.03160.

[213] Lyle Regenwetter, Rosen Yu, Cyril Picard, and Faez Ahmed. Engineering regression without real-data training: Domain adaptation for tabular foundation models using multi-dataset embeddings, 2026. URL https://arxiv.org/abs/2603.04692.

[214] Mohammad Hossein Nikzad, Mohammad Heidari-Rarani, and Pooya Sareh. From classical machine learning algorithms to modern transformer-inspired neural networks for multi-target prediction of fracture properties in concrete structures. *Machine Learning with Applications*, 24:100877, 2026. doi: 10.1016/j.mlwa.2026.100877. URL https://doi.org/10.1016/j.mlwa.2026.100877.

[215] Ke Wang, Yifan Tang, Nguyen Gia Hien Vu, Faez Ahmed, and G. Gary Wang. Tabpfn for zero-shot parametric engineering design generation, 2026. URL https://arxiv.org/abs/2602.02735.

[216] Qinyang Li, Rongzhi Dong, Nicholas Miklaucic, Jeffrey Hu, Sadman Sadeed Omee, Lai Wei, Sourin Dey, Ming Hu, and Jianjun Hu. In context learning foundation models for materials property prediction with small datasets, 2025. URL https://arxiv.org/abs/2601.00133.

[217] Taiga Saito, Yu Otake, and Stephen Wu. Applying a tabular foundation model to geotechnical site characterization. *Geodata and AI*, 6:100040, March 2026. ISSN 3050-483X. doi: 10.1016/j.geoai.2025.100040. URL http://dx.doi.org/10.1016/j.geoai.2025.100040.

[218] Jyothsna K, Malathi R, Rajalakshmi S, Preetha Achuthan, Senthil Kumar Muniasamy, and Lavanya S. Advanced visualization and interpretable machine learning for performance prediction of biochar-modified concrete. *E3S Web of Conferences*, 702:01008, 2026. doi: 10.1051/e3sconf/202670201008. URL https://doi.org/10.1051/e3sconf/202670201008.

[219] Tong Liu, Deji Xie, Tangzhi Liu, Jue Shan, and Caiqing Tang. Prediction of driver alertness levels on mountain roads using machine learning models: A naturalistic driving study in china. *Traffic Injury Prevention*, pages 1–10, 2025. doi: 10.1080/15389588.2025.2577155. URL https://doi.org/10.1080/15389588.2025.2577155.

[220] Wei-Tian Lu, Ze-Zhao Wang, and Xin-Yu Zhao. More trustworthy prediction of elastic modulus of recycled aggregate concrete using mcbe and tabpfn. *Materials*, 18:5221, 2025. doi: 10.3390/ma18225221. URL https://doi.org/10.3390/ma18225221.

[221] Yanjin Zhang and Zefeng Yu. Psf-net: Uncertainty-aware fusion of tabpfn and saint for 5g base-station electromagnetic radiation prediction. In *SAE Technical Paper Series*, volume 1. SAE International, 2025. doi: 10.4271/2025-99-0127. URL https://doi.org/10.4271/2025-99-0127.

[222] Wei Zhu, Na Xu, and James C. Hower. Demystifying hardgrove grindability index prediction using interpretable machine learning models. *Fuel*, 423:139297, 2026. doi: 10.1016/j.fuel.2026.139297. URL https://doi.org/10.1016/j.fuel.2026.139297.

[223] Jianglei Xing, Xiao Tan, Yihao Li, Dongzhao Jin, Pengwei Guo, Yuhuan Wang, and Huiya Niu. Interpretable machine learning for predicting splitting strength of asphalt concrete: Insights from shap analysis. Preprint at Preprints.org, 2026. URL https://doi.org/10.20944/preprints202603.2259.v1.

[224] Dong Wang, Feng Ju, and Go Igarashi. Cleaner production–oriented design of cemented foam backfill with high strength, low cost, and low carbon emissions: A tabpfn-based multi-objective optimization framework. *Journal of Cleaner Production*, 554:148119, 2026. doi: 10.1016/j.jclepro.2026.148119. URL https://doi.org/10.1016/j.jclepro.2026.148119.

[225] Liang Qin, Tong Liu, Qianhui Sun, and Mingxin Tang. An interpretable pretrained tabular modeling framework for predicting iri across multiple pavement structural configurations. *Buildings*, 16:1358, 2026. doi: 10.3390/buildings16071358. URL https://doi.org/10.3390/buildings16071358.

[226] Huanbao Zhang, Fengping Xu, Yu Yin, Linhai Wan, Jie Guo, Haiyang He, Qibin Lin, Shenchen Zhang, Shijiao Yang, and Fulin Wang. Strength prediction of cemented paste backfill with different machine learning and shapley additive explanation (shap) approaches. *Results in Engineering*, 28:108269, 2025. doi: 10.1016/j.rineng.2025.108269. URL https://doi.org/10.1016/j.rineng.2025.108269.

[227] Limao Zhang, Benyinan Huang, Yongsheng Li, Cheng Meng, and Maozhi Wu. Data-driven robust adverse geological conditions detection in tunnel construction considering uncertainty. *Advanced Engineering Informatics*, 74:104615, 2026. doi: 10.1016/j.aei.2026.104615. URL https://doi.org/10.1016/j.aei.2026.104615.

[228] Taiga Saito, Yu Otake, Daijiro Mizutani, and Stephen Wu. Tabpfn extensions for interpretable geotechnical modelling, 2026. URL https://arxiv.org/abs/2603.21033.

[229] Sergey A. Stel'makh, Alexey N. Beskopylny, Evgenii M. Shcherban', Irina Razveeva, Samson Oganesyan, Diana M. Shakhalieva, Andrei Chernil'nik, and Gleb Onore. Compressive strength of geopolymer concrete prediction using machine learning methods. *Algorithms*, 18:744, 2025. doi: 10.3390/a18120744. URL https://doi.org/10.3390/a18120744.

[230] Lin Deng, Linghui Xie, Sijia Zhu, Zhi Li, and Fangzhou Lin. Enhancing the prediction accuracy of concrete properties with knowledge constrained data augmentation and tabular foundation model. *Applied Soft Computing*, 195:115037, 2026. doi: 10.1016/j.asoc.2026.115037. URL https://doi.org/10.1016/j.asoc.2026.115037.

[231] Hang Yin, Jiawei Chen, Fan Guo, and Jiangang Yang. Surrogate-assisted multi-objective optimization of leaf-vein textured journal bearings under thermohydrodynamic lubrication. *Tribology International*, 220:111936, 2026. doi: 10.1016/j.triboint.2026.111936. URL https://doi.org/10.1016/j.triboint.2026.111936.

[232] Ya Mao, Yuhang Li, Yanhui Lai, and Fangshuo Fan. A data-driven reduced-order model for rotary kiln temperature field prediction using autoencoder and tabpfn. *Applied Sciences*, 16:2029, 2026. doi: 10.3390/app16042029. URL https://doi.org/10.3390/app16042029.

[233] Zhuohua Liu, Kaiqi Huang, Qinxin Mei, Yuanqi Hu, and Wei W. Xing. Exploiting function-family structure in analog circuit optimization, 2025. URL https://arxiv.org/abs/2512.00712.

[234] Nicolò Bellarmino, Riccardo Cantoro, Martin Huch, and Tobias Kilian. Minimal supervision, maximum accuracy: Tabpfn for microcontroller performance prediction. In *Proceedings of the International Test Conference (ITC)*, 2025. doi: 10.1109/ITC58126.2025.00067. URL https://iris.polito.it/handle/11583/3002056. Applies TabPFN for MCU performance screening with minimal supervision.

[235] Ping He, Zhanlin Cao, Honggui Di, Guangxin Shen, and Shunhua Zhou. Application of machine learning in caisson inclination prediction: Model performance comparison and interpretability analysis. *Underground Space*, 2025. URL https://www.sciencedirect.com/science/article/abs/pii/S2214391225001734. Includes TabPFN-based models among compared approaches.

[236] Zheyuan Lin, Xinhang Lin, Wanxin Li, Zhongxing Tian, Xudong Chai, Dongdong Zou, Weilin Xie, Yi Dong, and Yi Cai. Rapid few-shot tabular machine learning for $\phi$-otdr event classification. *Optics Express*, 33(17):36646–36662, 2025. doi: 10.1364/OE.571235. URL https://opg.optica.org/oe/fulltext.cfm?uri=oe-33-17-36646&id=575783.

[237] Sergio Ruiz-Villafranca, José Roldán-Gómez, Juan Manuel Castelo Gómez, Javier Carrillo-Mondéjar, and José Luis Martinez. A tabpfn-based intrusion detection system for the industrial internet of things. *The Journal of Supercomputing*, 80:20080–20117, 2024. doi: 10.1007/s11227-024-06166-x. URL https://doi.org/10.1007/s11227-024-06166-x.

[238] Zongzheng Li, Chunru Xiong, Kai Zheng, and Qiang Li. An rf-tabpfn-based framework for few-shot iot network attack recognition using lasso-rfe feature selection. *IEEE Access*, 13:151452–151465, 2025. URL https://ieeexplore.ieee.org/document/11142329. Combines Random Forest and TabPFN; DOI to be taken from the IEEE record.

[239] S. Chang and co authors. Cryogenic assisted abrasive waterjet machining of ti-6al-4v alloy: Thermo-mechanical optimization and ai-based surface integrity prediction. Article available via ScienceDirect, 2025. URL https://www.sciencedirect.com/science/article/abs/pii/S2214993725004531. Includes TabPFN-based modeling for surface integrity.

[240] Bichen Shang, Guanzhe Li, Wei Sun, Liang Zhang, et al. In-context learning for nano-pcm thermal behavior prediction in battery thermal management via lattice boltzmann simulation. *Energy*, 2025. URL https://www.sciencedirect.com/science/article/pii/S036054422504335X. Evaluates TabPFN-style in-context learning for nano-PCM thermal behavior.

[241] Jie Wang, Junqi Deng, Siyi Li, Weijie Du, Zengqi Zhang, and Xiaoming Liu. Explainable machine learning for multicomponent concrete: Predictive modeling and feature interaction insights. *Materials*, 18(19):4456, 2025. doi: 10.3390/ma18194456. URL https://www.mdpi.com/1996-1944/18/19/4456.

[242] Yongyong Jia, Xiaohui Gao, Zhihui Cai, Yafeng Ji, and Qiwei He. The multimodal fusion framework reveals the mapping relationship between microstructure and friction behavior. SSRN preprint 5616984, 2025. URL https://ssrn.com/abstract=5616984. Integrates image features with a TabPFN-based module for wear prediction.

[243] J. Xu and co authors. Multiscale prediction from ion concentrations to soil salinity in salinized farmland using machine learning. SSRN preprint 5591702, 2025. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5591702. Compares multiple models; TabPFN achieves strong performance for soil salinity prediction.

[244] Paul Hofman, Timo Löhr, Maximilian Muschalik, Yusuf Sale, and Eyke Hüllermeier. Efficient credal prediction through decalibration, 2026. URL https://arxiv.org/abs/2603.08495.

[245] Mohamed Salem. Valid feature-level inference for tabular foundation models via the conditional randomization test, 2026. URL https://arxiv.org/abs/2603.06609.

[246] Lars Henry Berge Olsen and Dennis Christensen. Computing conditional shapley values using tabular foundation models, 2026. URL https://arxiv.org/abs/2602.09489.

[247] Jeongwhan Choi, Woosung Kang, Minseo Kim, Jongwoo Kim, and Noseong Park. Can tabpfn compete with gnns for node classification via graph tabularization?, 2025. URL https://arxiv.org/abs/2512.08798.

[248] Yukun Du, Haiyue Yu, Xiaotong Xie, Yan Zheng, Lixin Zhan, Yudong Du, Chongshuang Hu, Boxuan Wang, and Jiang Jiang. Meta-black-box optimization with bi-space landscape analysis and dual-control mechanism for saea, 2025. URL https://arxiv.org/abs/2511.15551.

[249] Panpan Qi, Xuanpeng Xiao, Gongming Yu, Haitao Yang, and Qiang Hu. Systematic study on the $\alpha$-particle preformation factor in the theory of $\alpha$-decay based on the tabular prior-data fitted network (tabpfn), 2026. URL https://arxiv.org/abs/2511.14705.

[250] Kenyon Ng, Edwin Fong, David T. Frazier, Jeremias Knoblauch, and Susan Wei. Tabmgp: Martingale posterior with tabpfn, 2026. URL https://arxiv.org/abs/2510.25154.

[251] Filip Sabo, Michele Meroni, Maria Piles, Martin Claverie, Fanie Ferreira, Elna Van Den Berg, Francesco Collivignarelli, and Felix Rembold. From rows to yields: How foundation models for tabular data simplify crop yield prediction, 2025. URL https://arxiv.org/abs/2506.19046.

[252] Guiyan Jiang and Donghui Zhang. Mitigating urban-centric bias to address the rural eligibility discovery lag. *Land*, 15:535, 2026. doi: 10.3390/land15040535. URL https://doi.org/10.3390/land15040535.

[253] Xianfeng Hu, Dongfeng Han, Quan Qin, Yanhong Que, Han Wang, Donghan Feng, Rui Chen, Jinkui Duan, Yanpeng Li, and Feng Li. Coastal soil salinity inversion using uav multispectral imagery and an interpretable stacking algorithm. *Remote Sensing*, 18:671, 2026. doi: 10.3390/rs18050671. URL https://doi.org/10.3390/rs18050671.

[254] Joao Fonseca and Julia Stoyanovich. Explainerpfn: Towards tabular foundation models for model-free zero-shot feature importance estimations, 2026. URL https://arxiv.org/abs/2601.23068.

[255] GitHub - Avuii/AsteroidSafe: Web dashboard that ingests NASA NeoWs / JPL SBDB data and classifies Near-Earth Objects as Potentially Hazardous (PHA) using a pretrained tabular foundation model (TabPFN), deployed via ONNX Runtime in .NET. — github.com. https://github.com/{A}vuii/{A}steroid{S}afe, . [Accessed 11-05-2026].

[256] Valentin Leroy, Shuvalaxmi Dass, and Sharif Ullah. Memory-based malware detection under limited data conditions: A comparative evaluation of tabpfn and ensemble models, 2026. URL https://arxiv.org/abs/2601.07305.

[257] Alan Inglis, Fiona Doohan, Subramani Natarajan, Breige McNulty, Chris Elliott, Anne Nugent, Julie Meneely, Brett Greer, Stephen Kildea, Diana Bucur, Martin Danaher, Melissa Di Rocco, Lisa Black, Adam Gauley, Naoise McKenna, and Andrew Parnell. Predicting mycotoxin contamination in irish oats using deep and transfer learning, 2025. URL https://arxiv.org/abs/2512.22243.

[258] E. Oukacha and Y. Becherini. Towards a unified scheme of blazar evolution, 2025. URL https://arxiv.org/abs/2507.03088.

[259] Weiying Zhao and Natalia Efremova. Grapevine disease prediction using climate variables from multi-sensor remote sensing imagery via a transformer model, 2024. URL https://arxiv.org/abs/2406.07094.

[260] GitHub - sebhaan/TabPFGen: TabPFGen: Synthetic Tabular Data Generation with TabPFN — github.com. https://github.com/sebhaan/{T}ab{P}{F}{G}en, . [Accessed 11-05-2026].

[261] Giulia Perciballi, Federica Granese, Ahmad Fall, Farida Zehraoui, Edi Prifti, and Jean-Daniel Zucker. Adapting tabpfn for zero-inflated metagenomic data. In *Table Representation Learning Workshop at NeurIPS 2024*, 2024. URL https://openreview.net/forum?id=3I0bVvUj25.

[262] Eloy Peña-Asensio, Josep M. Trigo-Rodríguez, Jordi Sort, Jordi Ibáñez-Insa, and Albert Rimola. Machine learning applications on lunar meteorite minerals: From classification to mechanical properties prediction. *International Journal of Mining Science and Technology*, 34(9):1283–1292, 2024. doi: 10.1016/j.ijmst.2024.08.001. URL https://www.sciencedirect.com/science/article/pii/S2095268624001010.

[263] Bihui Lu, Kun Yu, Lin Qiu, Huayong Li, Hongxing Wang, Xiaohong Liu, Jie Shan, and Nan Li. Predicting county-level winter wheat yield in eastern china using multi-source spatiotemporal data: An explainable machine learning approach. SSRN preprint, 2025. URL https://ssrn.com/abstract=5380177.

[264] Melina Thegarza and co authors. Ml_climate final project: Flood impact on housing prices. Course project report, GitHub repository, 2023. URL https://github.com/melina-thegarza/ml-climate/blob/main/doc/ML_Climate___Final.pdf. Student project using machine learning (incl. TabPFN) for flood impact assessment.

[265] Viacheslav Barkov, Jonas Schmidinger, Robin Gebbers, and Martin Atzmüller. Modern neural networks for small tabular datasets: The new default for field-scale digital soil mapping? *arXiv preprint arXiv:2508.09888*, 2025. URL https://arxiv.org/abs/2508.09888.

[266] Xiangang Zhu, Peidong Su, Jiang Yu, Jiaheng Pei, Zhaoyong Teng, Yougui Li, and Yuxuan Liu. A prediction model for hazard levels of shallow natural gas in tunnel based on k-means clustering and tabular prior-data fitted network. *Results in Engineering*, 21:106873, 2025. doi: 10.1016/j.rineng.2025.106873. URL https://www.sciencedirect.com/science/article/pii/S2590123025029366.

[267] Nasser Alkhulaifi and Nicholas Bowler. Autoenergy: An automated feature engineering algorithm for energy consumption forecasting with automl. *Knowledge-Based Systems*, 2025. URL https://www.sciencedirect.com/science/article/pii/S0950705125013413. Early access; uses AutoML including TabPFN among evaluated models.

[268] Sunil Kumar Jha, James Brinkhoff, Andrew J. Robson, and Brian W. Dunn. Integrating remote sensing and weather time series for australian irrigated rice phenology prediction. *Remote Sensing*, 17 (17):3050, 2025. doi: 10.3390/rs17173050. URL https://www.mdpi.com/2072-4292/17/17/3050.

[269] Olanrewaju Daramola, Emmanuel Olanrewaju, Israel Trejo, and Elvis Enebeli. A target-specific machine learning framework for predicting fuel blend properties. ChemRxiv preprint, 2025. URL https://chemrxiv.org/engage/chemrxiv/article-details/68dc888d3e708a7649ff0ec9.

[270] Jonas Schmidinger, Viacheslav Barkov, Sebastian Vogel, Martin Atzmüller, and Gerard B. M. Heuvelink. Kriging prior regression: A case for kriging-based spatial features with tabpfn in soil mapping. *arXiv preprint arXiv:2509.09408*, 2025. URL https://arxiv.org/abs/2509.09408.

[271] Nikunj Panchal, Abdul Qayum, Abdul Shahid, et al. Metrics-first, language-aware clone type recognition: Auditable signals across c, c#, java, and python. *Authorea Preprints*, 2025. doi: 10.22541/au.176059643.37779565/v1. URL https://wiley.authorea.com/users/980519/articles/1346750-metrics-first-language-aware-clone-type-recognition-auditable-signals-across-c-c-java-

[272] Jacob Feitelberg, Dwaipayan Saha, Kyuseong Choi, Zaid Ahmad, Anish Agarwal, and Raaz Dwivedi. Tabimpute: Accurate and fast zero-shot missing-data imputation with a pre-trained transformer. *arXiv preprint arXiv:2510.02625*, 2025. doi: 10.48550/arXiv.2510.02625. URL https://arxiv.org/abs/2510.02625.

[273] Pablo García, Jordi de Curtò, Ignacio de Zarzà, Juan Carlos Cano, and Carlos T. Calafate. Foundation models for cybersecurity: A comprehensive multi-modal evaluation of tabpfn and tabicl for tabular intrusion detection. *Electronics*, 14(19):3792, 2025. doi: 10.3390/electronics14193792. URL https://www.mdpi.com/2079-9292/14/19/3792.

[274] Carola Sophia Heinzel, Lennart Purucker, Frank Hutter, and Peter Pfaffelhuber. Advancing biogeographical ancestry predictions through machine learning. In *Forensic Science International: Genetics*. Elsevier, 2025. doi: 10.1016/j.fsigen.2025.103290.

[275] Muhammad Moshiur Rahman, Andrew Robson, and Theo Bekker. Machine learning approaches for assessing avocado alternate bearing using sentinel-2 and climate variables—a case study in limpopo, south africa. *Preprints*, 2025(202510.2413), 2025. doi: 10.20944/preprints202510.2413.v1. URL https://www.preprints.org/manuscript/202510.2413. Preprint, version 1.

# Appendix Table of Contents

# A Contributors

**Model Development & Deployment.** Noah Hollmann, Frank Hutter, Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Mihir Manium, Shi Bin (Liam) 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

**Distribution & Product.** Sauraj Gambhir, Clara Cornu, Lilly Charlotte Wehrhahn, Diana Kriuchkova

**Operations.** Kursat Kaya, Lydia Sidhoum, Marie Salmon, Jerry Chen

*Authors are ordered by their date of joining Prior Labs; all authors above affiliated with Prior Labs at the time of contribution; work done at Prior Labs.*

**Scientific Advisors.** Samuel Müller, Madelon Hulsebos, Yann LeCun, Bernhard Schölkopf

*Scientific advisors did not contribute IP.*

# B Acknowledgements

We acknowledge the EuroHPC Joint Undertaking for awarding this project access to the EuroHPC supercomputer LUMI, hosted by CSC (Finland) and the LUMI consortium through a EuroHPC Regular Access call.


# C Architectural Hyperparameters

The tables below list the architectural hyperparameters of the released TabPFN-3 classifier and regressor checkpoints. The two models share all hyperparameters; the only differences are in the output decoder, which is task-specific (noted where applicable).

**Table 2.** Stage 1 — Feature embedding.

| Hyperparameter | Value | Description |
|---|---|---|
| `embed_dim` | 128 | Base embedding dimension used throughout the model |
| `feature_group_size` | 3 | Features per circular-shift group |
| `dist_embed_num_blocks` | 3 | Induced self-attention blocks |
| `dist_embed_num_heads` | 8 | Attention heads per block |
| `dist_embed_num_inducing_points` | 128 | Inducing points per column |

**Table 3.** Stage 2 — Feature aggregation.

| Hyperparameter | Value | Description |
|---|---|---|
| `feat_agg_num_blocks` | 3 | Transformer blocks |
| `feat_agg_num_heads` | 8 | Attention heads per block |
| `feat_agg_num_cls_tokens` | 4 | CLS tokens aggregated per row |
| `use_rope` | True | Rotary positional embeddings (RoPE) enabled |
| `feat_agg_rope_base` | 100 000 | RoPE base frequency $\theta$ |

**Table 4.** Stage 3 — ICL transformer.

| Hyperparameter | Value | Description |
|---|---|---|
| `icl_emsize` (derived) | 512 | `embed_dim` $\times$ `feat_agg_num_cls_tokens` $= 128 \times 4$ |
| `nlayers` | 24 | Transformer blocks |
| `icl_num_heads` | 8 | Query heads per block |
| `icl_num_kv_heads` | 8 | KV heads for train rows (standard MHA) |
| `icl_num_kv_heads_test` | 1 | KV heads for test rows |

**Table 5.** Many class output decoder — classifier.

| Hyperparameter | Value | Description |
|---|---|---|
| `max_num_classes` | 160 | Maximum supported class count |
| `decoder_num_heads` | 6 | Attention heads in retrieval decoder |
| `decoder_head_dim` | 64 | Head dimension in retrieval decoder |

**Table 6.** MLP output decoder — regressor (2-layer MLP).

| Hyperparameter | Value | Description |
|---|---|---|
| `architecture` (derived) | $512 \to 1024 \to 5000$ | `icl_emsize` $\to$ `icl_emsize` $\times$ `ff_factor` $\xrightarrow{\text{GELU}}$ `num_buckets` |
| `num_buckets` | 5000 | Output buckets for quantile regression |

**Table 7.** Shared settings (both classifier and regressor).

| Hyperparameter | Value | Description |
|---|---|---|
| `ff_factor` | 2 | Feed-forward expansion factor (all stages) |
| `softmax_scaling_mlp_hidden_dim` | 64 | Hidden units in query-aware softmax-scaling MLPs |

# D Prior visualizations

We provide a number of illustrative visualizations for the improvements to our prior. Figure 23 shows directed acyclic graphs sampled by our new graph-sampling algorithms; Figure 24 visualizes the functional relationships generated by the new combiner mechanisms; Figure 25 gives an example classification dataset generated from the prior; and Figure 26 demonstrates TabPFN-3's extrapolation capabilities, comparing to CatBoost.

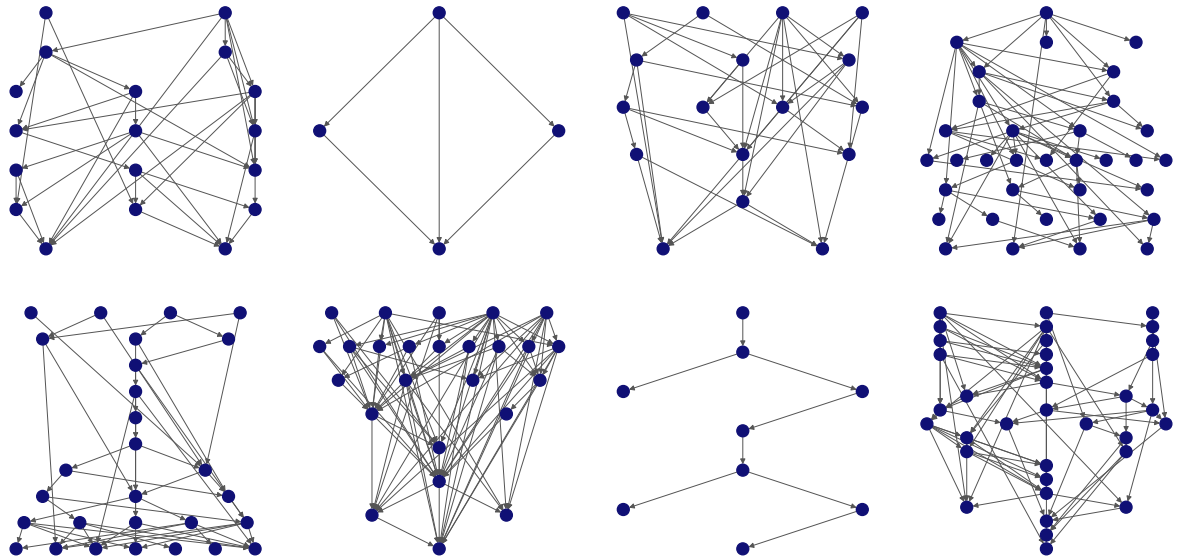


**Figure 23. Visualization of directed acyclic graphs underlying our SCM prior**, produced by our new graph sampling algorithms.

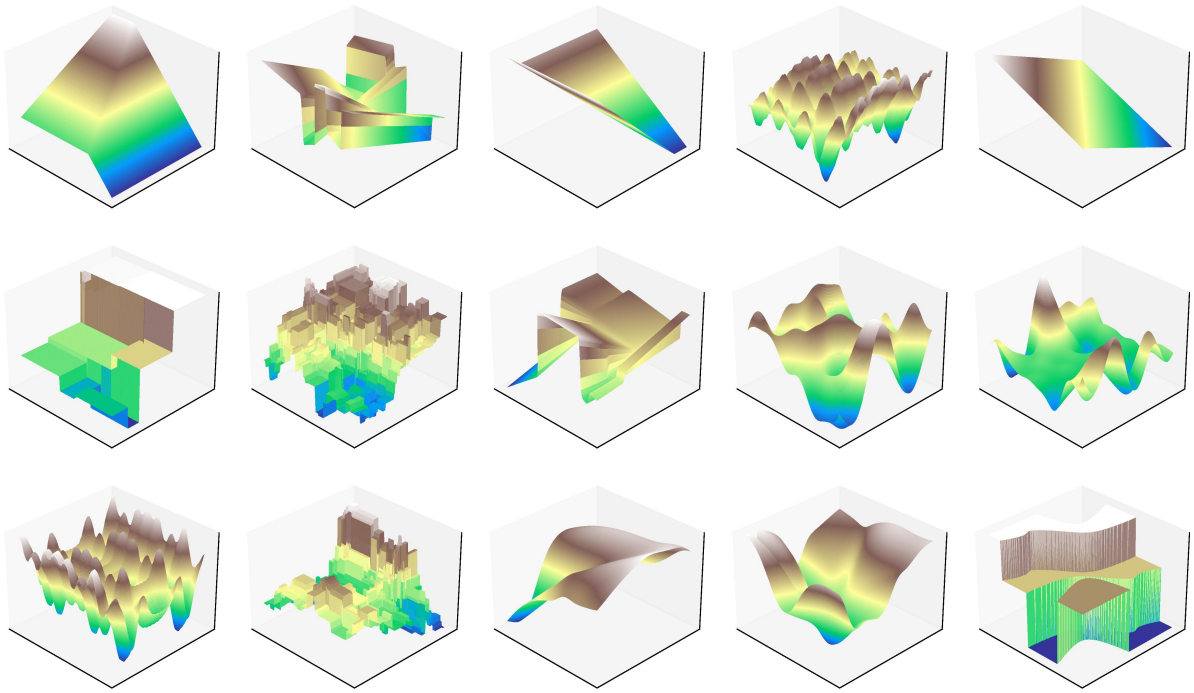


**Figure 24. Visualization of functional relationships generated by the new combiner mechanisms in our SCM prior.** While mechanisms in the prior have variable dimensionality, for the sake of visualization we plot functions on a two-dimensional grid.

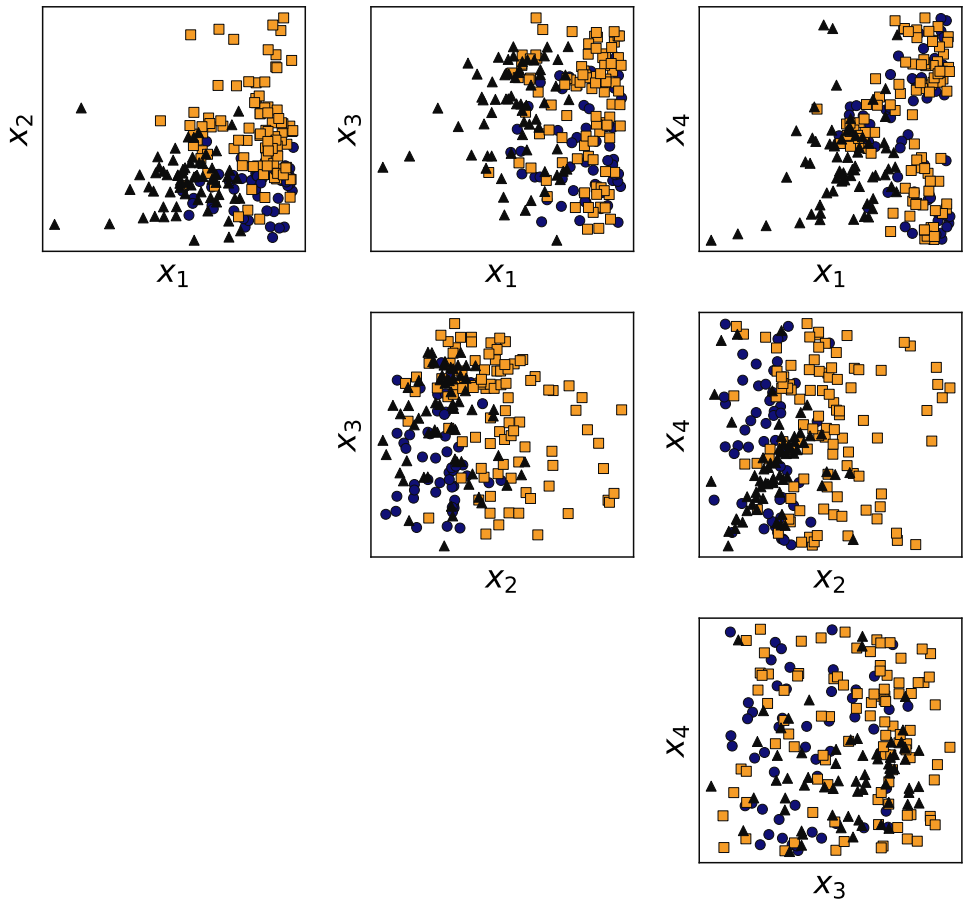



**Figure 25. Example classification dataset generated from the prior.** There are four covariates and the subplot in row i column j corresponds to a scatter plot of covariates i and j+1 with target class indicated by color.

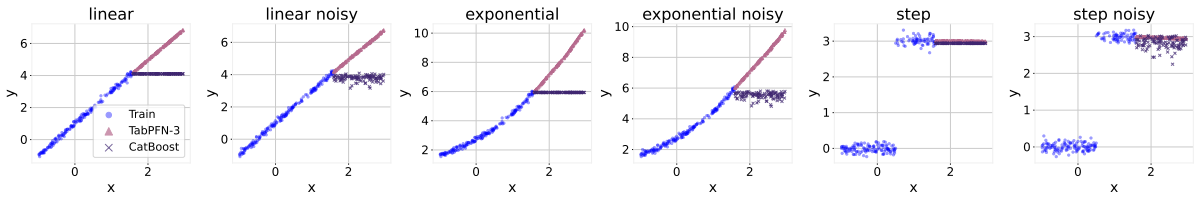



**Figure 26. Example demonstrating the extrapolation capabilities of TabPFN-3 (using our out-of-distribution compatible preprocessing), comparing to CatBoost.** As can be seen, TabPFN-3 is able to extrapolate successfully, which tree-based algorithms and tabular foundation models often struggle with.

# E Experimental results details

## E.1 Details on Causal Inference Results

**Causal Inference.** Many practical problems are rooted in causal logic, requiring an understanding of how interventions, rather than mere associations, shape outcomes. Estimating Conditional Average Treatment Effects (CATEs) serves as a primary tool for addressing these "what-if" scenarios, quantifying the expected change in an individual's response when a treatment is applied compared to when it is withheld. Previous results [18] have shown that TabPFN-2.5, especially when used as a T-Learner [65], achieves SOTA performance on the RealCause benchmark [67]. While TabPFN-3 does not quite achieve the highest performance on RealCause (still surpassed by TabPFN-2.5), we see substantial improvements on larger datasets with up to 50k samples in the `scikit-uplift` library [66]. We describe the details of

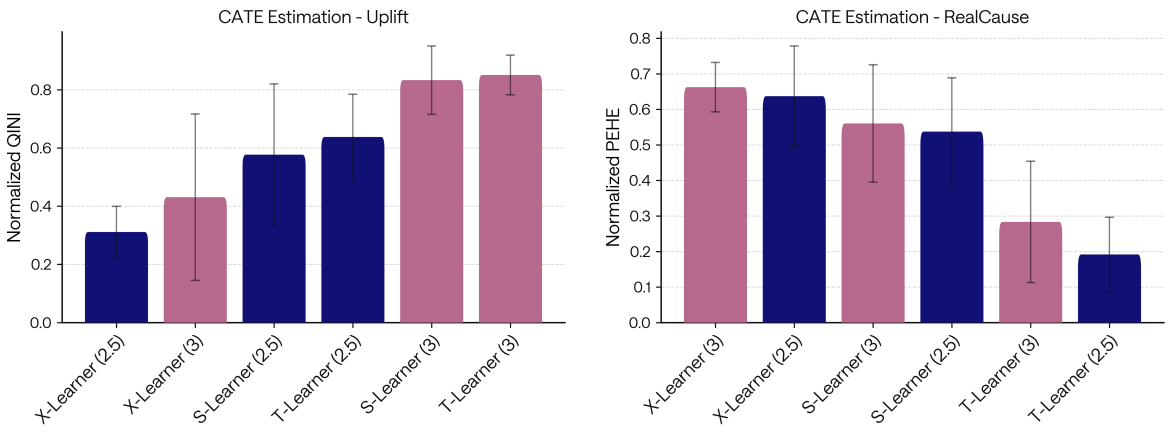



**Figure 27. TabPFN-3 as a T/X/S-Learner.** TabPFN-3 when used as a T/S-Learner achieves strong performance in terms of QINI-score (↑) in Uplift Modeling on the `scikit-uplift` benchmark. We report worsened performance in terms of PEHE (↓) on the RealCause benchmark compared to the previous version.

this evaluation below.

**Real-World QINI Evaluation.** One of the major drawbacks in evaluating causal inference methods is referred to by Holland [73] as the Fundamental Problem of Causal Inference, which states that individual treatment effects can never actually be observed in the real-world. In simple terms, one cannot experimentally test both potential outcomes without interference. Under the assumption of experimental (RCT) data, Uplift Modeling [66] allows to evaluate the benefit of using a causal estimator in terms of ranking individuals by treatment effect. Crucially, this evaluation strategy does not require access to ground truth (synthetic) treatment effects, and serves as arguably the most real-world evaluation of CATE estimators, for example, when A/B testing data is available. Using only observed treatment and outcomes, one can compute the Area-Under the QINI Curve (AUC-QINI) to evaluate CATE estimators by their ability to identify individuals for which the treatment has a strong impact.

**Strong Performance in Uplift Modeling.** We report the mean normalized AUC-QINI score for the T/X/S meta-learners using TabPFN-2.5 and 3 (Figure 27). TabPFN-3 used as an S and T-Learner achieves stronger performance than other baselines. We observe somewhat worsened performance on the RealCause benchmark [67], which is characterized by smaller sample sizes.

## E.2 Detailed TabArena Results

### E.2.1 Evaluation Metrics

We re-use the official TabArena [1] evaluation metrics and code for generating TabArena plots and tables.

**Elo**: Following TabArena, we evaluate models using the Elo rating system [74]. Elo is a pairwise comparison-based rating system where each model's rating predicts its expected win probability against others, with a 400-point Elo gap corresponding to a 10 to 1 (91%) expected win rate. We calibrate 1000 Elo to the performance of the default TabArena random forest configuration across all figures, and perform 200 rounds of bootstrapping to obtain 95% confidence intervals, similar to what is done in ChatBot Arena [75]. In our TabArena results, Elo scores are computed using ROC AUC for binary classification, log-loss for multiclass classification, and RMSE for regression.

**Improvability**: The improvability metric introduced in TabArena measures how many percent lower the error of the best method is than the current method on a dataset. This is then averaged over datasets. Formally, for a single dataset,

$$\text{Improvability} := \frac{\text{err}_i - \text{best\_err}_i}{\text{err}_i} \cdot 100\% \ .$$

Improvability is always between 0% and 100%.

### E.2.2 Experiment Details

For all TabArena results, we run experiments using the official TabArena code and evaluation pipeline. We will contribute a reproducible official TabArena submission for TabPFN-3 shortly after it becomes publicly available. While not strictly necessary to make predictions on test data, we follow TabArena's fit time procedure of fitting an 8-fold bagged ensemble to generate a cross-validation score followed by refitting the model on the full training data at test time, as is done for the other tabular foundation models on TabArena.

All results for non-TabPFN-3 models in our TabArena experiments were from the official TabArena reported results. All cached results from tabular foundation models (TabPFN-2.5, TabPFN-2.6, TabICLv2 and TabDPT) were run on a single H200 GPU, while all results for TabPFN-3 and TabPFN-3-Plus (Thinking) were run on a single RTX 6000 GPU, a weaker GPU compared to an H200.

For both TabPFN-3 and TabPFN-3-Plus (Thinking), we ran all splits of TabArena, which includes a total of 816 tasks across 51 datasets. In all cases we report results for all splits of each dataset.

### E.2.3 TabArena Pareto Frontier Explanation

Figure 2 and Figure 11 show TabArena Pareto frontiers of models across Improvability and the median combined train + inference time per 1000 samples. The connected points for a given model type indicate tuning + ensembling performance with points from left to right marking ensembles of increasing numbers of random configurations (1, 2, 5, 10, 25, 50, 100, 150, 201). The trajectories are sampled 20 times from all trials and averaged. The left-most points use the default configuration, and the right-most highlighted points use all configurations.

### E.2.4 TabArena Leaderboard Tables

We present the leaderboard tables for TabArena, TabArena-medium, TabArena-small, TabArena-classification, and TabArena-regression, below.

For all 5 views, TabPFN-3 ranks highest among all models on TabArena, while TabPFN-3-Plus (Thinking) pushes even futher, strongly outperforming AutoGluon 1.5 extreme and ranking first in Elo, wins, and Improvability in every leaderboard.

**Table 8. TabArena leaderboard using all 51 datasets with 816 total tasks.**

| Model | Elo (↑) | #wins (↑) | Improvability (↓) | Train time per 1K [s] | Predict time per 1K [s] |
|---|---|---|---|---|---|
| TabPFN-3-Thinking | **1800**$_{-72,+105}$ | **13.2** | **4.7%** | 37.69 | 3.26 |
| AutoGluon 1.5 (extreme, 4h) | **1695**$_{-68,+83}$ | **5.8** | **5.7%** | 289.07 | 4.03 |
| TabPFN-3 (D) | **1677**$_{-62,+86}$ | **6.3** | **6.9%** | 2.31 | 0.74 |
| TabPFN-2.6 (D) | 1623$_{-56,+78}$ | 1.3 | 8.7% | 5.48 | 0.55 |
| RealTabPFN-2.5 (T+E) | 1602$_{-62,+79}$ | 2.1 | 8.3% | 2040.22 | 8.92 |
| TabICLv2 (D) | 1599$_{-64,+77}$ | 5.3 | 7.7% | 4.02 | 0.38 |
| RealTabPFN-2.5 (T) | 1559$_{-56,+69}$ | 1.4 | 9.1% | 2040.22 | 1.22 |
| RealTabPFN-2.5 (D) | 1526$_{-48,+66}$ | 0.9 | 9.5% | 5.81 | 0.64 |
| RealMLP (T+E) | 1514$_{-45,+58}$ | 0.5 | 11.2% | 2950.72 | 11.99 |
| TabDPT (T+E) | 1461$_{-54,+63}$ | 2.0 | 11.7% | 4907.64 | 286.65 |
| TabM (T+E) | 1449$_{-44,+56}$ | 1.0 | 12.6% | 3285.87 | 1.47 |
| LightGBM (T+E) | 1438$_{-31,+36}$ | 0.1 | 13.6% | 416.98 | 2.64 |
| RealMLP (T) | 1433$_{-47,+48}$ | 0.4 | 12.5% | 2950.72 | 0.66 |
| CatBoost (T+E) | 1420$_{-42,+41}$ | 0.1 | 13.2% | 1658.41 | 0.65 |
| CatBoost (T) | 1410$_{-45,+41}$ | 0.5 | 13.4% | 1658.41 | 0.08 |
| TabDPT (T) | 1405$_{-56,+60}$ | 0.7 | 12.9% | 4907.64 | 39.96 |
| TabM (T) | 1392$_{-43,+54}$ | 0.3 | 13.5% | 3285.87 | 0.17 |
| LightGBM (T) | 1390$_{-29,+33}$ | 0.0 | 14.3% | 416.98 | 0.33 |
| XGBoost (T+E) | 1379$_{-35,+34}$ | 0.1 | 14.4% | 693.49 | 1.69 |
| CatBoost (D) | 1371$_{-44,+40}$ | 0.2 | 14.2% | 6.83 | 0.08 |
| XGBoost (T) | 1354$_{-35,+33}$ | 0.0 | 14.7% | 693.49 | 0.31 |
| TabDPT (D) | 1326$_{-56,+68}$ | 0.3 | 15.3% | 47.62 | 43.74 |
| TabM (D) | 1299$_{-44,+49}$ | 0.2 | 15.7% | 10.49 | 0.13 |
| RealMLP (D) | 1234$_{-37,+38}$ | 0.1 | 17.1% | 10.06 | 1.69 |
| XGBoost (D) | 1215$_{-38,+39}$ | 0.0 | 17.5% | 1.94 | 0.12 |
| LightGBM (D) | 1189$_{-29,+34}$ | 0.0 | 18.0% | 1.96 | 0.14 |

**Table 9. TabArena-medium leaderboard on the 15 largest datasets in TabArena**, with 10k–100k training samples, evaluated on the full 135 tasks with 9 splits per dataset.

| Model | Elo (↑) | #wins (↑) | Improvability (↓) | Train time per 1K [s] | Predict time per 1K [s] |
|---|---|---|---|---|---|
| TabPFN-3-Thinking | **2146**$_{-87,+121}$ | **6.2** | **1.3%** | 15.10 | 2.15 |
| AutoGluon 1.5 (extreme, 4h) | **1907**$_{-50,+92}$ | 1.4 | **3.4%** | 191.18 | 2.21 |
| TabPFN-3 (D) | **1835**$_{-137,+224}$ | **3.3** | **4.1%** | 0.83 | 0.27 |
| TabPFN-2.6 (D) | 1741$_{-72,+121}$ | 0.0 | 6.4% | 2.76 | 0.70 |
| TabICLv2 (D) | 1712$_{-108,+208}$ | **1.9** | 5.3% | 0.76 | 0.14 |
| RealTabPFN-2.5 (T+E) | 1663$_{-111,+149}$ | 0.0 | 7.2% | 735.58 | 11.74 |
| RealMLP (T+E) | 1645$_{-94,+91}$ | 0.0 | 7.4% | 1719.82 | 1.67 |
| CatBoost (T+E) | 1625$_{-64,+86}$ | 0.0 | 7.4% | 777.59 | 0.25 |
| CatBoost (T) | 1616$_{-67,+95}$ | 0.3 | 7.6% | 777.59 | 0.05 |
| RealTabPFN-2.5 (T) | 1612$_{-103,+130}$ | 0.1 | 7.9% | 735.58 | 1.39 |
| LightGBM (T+E) | 1604$_{-56,+70}$ | 0.0 | 9.2% | 131.56 | 2.64 |
| CatBoost (D) | 1576$_{-106,+105}$ | 0.1 | 7.8% | 3.24 | 0.03 |
| XGBoost (T+E) | 1565$_{-61,+90}$ | 0.1 | 9.3% | 282.13 | 0.56 |
| RealMLP (T) | 1554$_{-85,+106}$ | 0.0 | 8.7% | 1719.82 | 0.08 |
| TabM (T+E) | 1538$_{-90,+157}$ | 0.7 | 9.1% | 1993.14 | 0.62 |
| RealTabPFN-2.5 (D) | 1536$_{-90,+141}$ | 0.0 | 8.7% | 1.88 | 0.64 |
| TabDPT (T+E) | 1533$_{-124,+142}$ | 0.8 | 8.8% | 4786.55 | 444.54 |
| LightGBM (T) | 1515$_{-59,+80}$ | 0.0 | 10.3% | 131.56 | 0.13 |
| XGBoost (T) | 1514$_{-56,+69}$ | 0.0 | 9.8% | 282.13 | 0.07 |
| TabM (T) | 1489$_{-90,+158}$ | 0.0 | 9.9% | 1993.14 | 0.06 |
| TabDPT (T) | 1411$_{-125,+121}$ | 0.0 | 11.3% | 4786.55 | 42.64 |
| XGBoost (D) | 1375$_{-115,+101}$ | 0.0 | 11.7% | 0.49 | 0.05 |
| TabDPT (D) | 1336$_{-144,+131}$ | 0.0 | 14.0% | 46.62 | 43.74 |
| TabM (D) | 1330$_{-101,+123}$ | 0.0 | 12.6% | 5.16 | 0.07 |
| RealMLP (D) | 1280$_{-71,+79}$ | 0.0 | 13.7% | 6.75 | 0.23 |
| LightGBM (D) | 1263$_{-63,+55}$ | 0.0 | 13.5% | 0.29 | 0.04 |

**Table 10. TabArena-small leaderboard on the 36 smallest datasets in TabArena**, with 500–10k training samples, evaluated on the full 681 tasks.

| Model | Elo (↑) | #wins (↑) | Improvability (↓) | Train time per 1K [s] | Predict time per 1K [s] |
|---|---|---|---|---|---|
| TabPFN-3-Thinking | **1723**$_{-60,+100}$ | **7.0** | **6.1%** | 52.78 | 3.40 |
| AutoGluon 1.5 (extreme, 4h) | **1641**$_{-57,+79}$ | **4.4** | **6.6%** | 346.57 | 6.56 |
| TabPFN-3 (D) | **1638**$_{-58,+85}$ | 2.9 | **8.1%** | 4.84 | 1.54 |
| RealTabPFN-2.5 (T+E) | 1598$_{-64,+97}$ | 2.1 | 8.7% | 2289.05 | 8.05 |
| TabPFN-2.6 (D) | 1596$_{-49,+74}$ | 1.3 | 9.7% | 7.03 | 0.55 |
| TabICLv2 (D) | 1574$_{-83,+105}$ | **3.4** | 8.7% | 7.06 | 0.67 |
| RealTabPFN-2.5 (T) | 1556$_{-58,+75}$ | 1.2 | 9.5% | 2289.05 | 1.14 |
| RealTabPFN-2.5 (D) | 1542$_{-52,+83}$ | 0.9 | 9.9% | 6.76 | 0.64 |
| RealMLP (T+E) | 1482$_{-47,+63}$ | 0.5 | 12.7% | 3770.75 | 21.90 |
| TabDPT (T+E) | 1448$_{-59,+76}$ | 1.2 | 12.9% | 5119.36 | 218.71 |
| TabM (T+E) | 1430$_{-52,+57}$ | 0.4 | 14.0% | 3553.12 | 1.74 |
| TabDPT (T) | 1414$_{-60,+72}$ | 0.7 | 13.6% | 5119.36 | 28.35 |
| RealMLP (T) | 1402$_{-42,+55}$ | 0.4 | 14.2% | 3770.75 | 1.78 |
| LightGBM (T+E) | 1392$_{-34,+37}$ | 0.1 | 15.5% | 892.41 | 2.57 |
| TabM (T) | 1368$_{-53,+56}$ | 0.3 | 15.0% | 3553.12 | 0.24 |
| CatBoost (T+E) | 1362$_{-43,+46}$ | 0.1 | 15.6% | 2476.51 | 0.81 |
| LightGBM (T) | 1357$_{-30,+36}$ | 0.0 | 15.9% | 892.41 | 0.35 |
| CatBoost (T) | 1351$_{-35,+48}$ | 0.1 | 15.8% | 2476.51 | 0.10 |
| TabDPT (D) | 1331$_{-67,+74}$ | 0.3 | 15.9% | 50.32 | 43.71 |
| XGBoost (T+E) | 1326$_{-37,+34}$ | 0.0 | 16.5% | 884.18 | 2.37 |
| CatBoost (D) | 1312$_{-35,+35}$ | 0.1 | 16.9% | 9.64 | 0.13 |
| XGBoost (T) | 1309$_{-39,+32}$ | 0.0 | 16.7% | 884.18 | 0.39 |
| TabM (D) | 1296$_{-47,+54}$ | 0.2 | 17.0% | 13.18 | 0.17 |
| RealMLP (D) | 1224$_{-42,+37}$ | 0.1 | 18.5% | 15.69 | 4.69 |
| LightGBM (D) | 1169$_{-40,+42}$ | 0.0 | 19.9% | 3.61 | 0.17 |
| XGBoost (D) | 1165$_{-38,+30}$ | 0.0 | 19.9% | 3.29 | 0.25 |

**Table 11. TabArena-classification leaderboard on the 38 classification datasets in TabArena.**

| Model | Elo (↑) | #wins (↑) | Improvability (↓) | Train time per 1K [s] | Predict time per 1K [s] |
|---|---|---|---|---|---|
| TabPFN-3-Thinking | **1782**$_{-72,+109}$ | **10.0** | **6.0%** | 35.70 | 3.00 |
| AutoGluon 1.5 (extreme, 4h) | **1689**$_{-82,+96}$ | **4.8** | **6.5%** | 267.31 | 3.98 |
| TabPFN-3 (D) | **1660**$_{-75,+91}$ | 3.7 | **8.7%** | 2.43 | 0.75 |
| TabPFN-2.6 (D) | 1604$_{-69,+69}$ | 0.5 | 10.6% | 5.17 | 0.54 |
| TabICLv2 (D) | 1593$_{-75,+94}$ | **4.1** | 9.3% | 4.15 | 0.41 |
| RealTabPFN-2.5 (T+E) | 1578$_{-75,+76}$ | 1.7 | 10.2% | 2046.25 | 8.98 |
| RealTabPFN-2.5 (T) | 1554$_{-66,+72}$ | 1.2 | 11.0% | 2046.25 | 1.33 |
| RealTabPFN-2.5 (D) | 1539$_{-63,+69}$ | 0.9 | 11.2% | 5.76 | 0.79 |
| RealMLP (T+E) | 1492$_{-45,+63}$ | 0.3 | 13.5% | 2879.46 | 12.49 |
| TabM (T+E) | 1464$_{-48,+75}$ | 1.0 | 14.8% | 2466.21 | 1.50 |
| LightGBM (T+E) | 1436$_{-37,+48}$ | 0.1 | 15.7% | 382.05 | 1.49 |
| RealMLP (T) | 1413$_{-47,+55}$ | 0.4 | 15.0% | 2879.46 | 0.60 |
| CatBoost (T+E) | 1412$_{-47,+55}$ | 0.1 | 15.2% | 1372.94 | 0.56 |
| TabM (T) | 1411$_{-58,+71}$ | 0.3 | 15.6% | 2466.21 | 0.18 |
| TabDPT (T+E) | 1411$_{-56,+80}$ | 0.5 | 14.5% | 4940.61 | 307.75 |
| CatBoost (T) | 1404$_{-45,+54}$ | 0.4 | 15.4% | 1372.94 | 0.07 |
| LightGBM (T) | 1392$_{-33,+43}$ | 0.0 | 16.4% | 382.05 | 0.25 |
| XGBoost (T+E) | 1382$_{-48,+50}$ | 0.1 | 16.5% | 685.87 | 1.45 |
| CatBoost (D) | 1381$_{-46,+46}$ | 0.2 | 16.0% | 5.72 | 0.08 |
| XGBoost (T) | 1356$_{-40,+45}$ | 0.0 | 16.8% | 685.87 | 0.21 |
| TabDPT (T) | 1351$_{-58,+66}$ | 0.6 | 16.0% | 4940.61 | 41.61 |
| TabM (D) | 1315$_{-48,+56}$ | 0.2 | 18.0% | 10.21 | 0.14 |
| TabDPT (D) | 1270$_{-57,+62}$ | 0.3 | 18.9% | 49.21 | 43.82 |
| RealMLP (D) | 1244$_{-34,+39}$ | 0.1 | 19.6% | 10.47 | 1.71 |
| XGBoost (D) | 1231$_{-50,+47}$ | 0.0 | 19.6% | 1.77 | 0.12 |
| LightGBM (D) | 1192$_{-40,+49}$ | 0.0 | 20.6% | 1.79 | 0.12 |

**Table 12. TabArena-regression leaderboard on the 13 regression datasets in TabArena.**

| Model | Elo (↑) | #wins (↑) | Improvability (↓) | Train time per 1K [s] | Predict time per 1K [s] |
|---|---|---|---|---|---|
| TabPFN-3-Thinking | **1959**$_{-150,+211}$ | **3.2** | **0.9%** | 43.00 | 3.26 |
| TabPFN-3 (D) | **1827**$_{-142,+255}$ | **2.5** | **1.6%** | 1.69 | 0.57 |
| AutoGluon 1.5 (extreme, 4h) | **1804**$_{-97,+133}$ | 1.1 | 3.2% | 335.03 | 4.33 |
| TabPFN-2.6 (D) | 1776$_{-71,+131}$ | 0.8 | 3.3% | 8.52 | 0.70 |
| RealTabPFN-2.5 (T+E) | 1774$_{-107,+174}$ | 0.5 | **2.6%** | 1709.05 | 8.12 |
| TabDPT (T+E) | 1748$_{-92,+171}$ | **1.5** | 3.5% | 4786.55 | 239.54 |
| TabICLv2 (D) | 1700$_{-159,+293}$ | 1.2 | 3.2% | 2.10 | 0.25 |
| TabDPT (T) | 1696$_{-79,+134}$ | 0.1 | 3.9% | 4786.55 | 38.50 |
| RealMLP (T+E) | 1677$_{-68,+126}$ | 0.2 | 4.3% | 3995.01 | 10.05 |
| RealTabPFN-2.5 (T) | 1654$_{-113,+165}$ | 0.2 | 3.4% | 1709.05 | 0.81 |
| TabDPT (D) | 1604$_{-72,+153}$ | 0.0 | 4.9% | 46.62 | 39.21 |
| RealMLP (T) | 1574$_{-84,+114}$ | 0.0 | 5.3% | 3995.01 | 0.84 |
| RealTabPFN-2.5 (D) | 1558$_{-110,+159}$ | 0.0 | 4.9% | 7.04 | 0.51 |
| CatBoost (T+E) | 1513$_{-73,+113}$ | 0.0 | 7.3% | 3552.96 | 0.97 |
| LightGBM (T+E) | 1509$_{-90,+107}$ | 0.0 | 7.7% | 700.15 | 9.32 |
| CatBoost (T) | 1489$_{-78,+119}$ | 0.1 | 7.4% | 3552.96 | 0.10 |
| TabM (T+E) | 1463$_{-96,+147}$ | 0.0 | 6.2% | 4158.29 | 1.41 |
| LightGBM (T) | 1440$_{-77,+119}$ | 0.0 | 8.3% | 700.15 | 0.97 |
| XGBoost (T+E) | 1424$_{-52,+72}$ | 0.0 | 8.2% | 834.93 | 2.61 |
| XGBoost (T) | 1403$_{-60,+85}$ | 0.0 | 8.4% | 834.93 | 0.39 |
| CatBoost (D) | 1389$_{-92,+107}$ | 0.0 | 8.9% | 10.89 | 0.09 |
| TabM (T) | 1381$_{-101,+147}$ | 0.0 | 7.1% | 4158.29 | 0.17 |
| TabM (D) | 1284$_{-118,+126}$ | 0.0 | 8.8% | 13.32 | 0.13 |
| RealMLP (D) | 1235$_{-81,+105}$ | 0.0 | 9.8% | 8.90 | 1.64 |
| LightGBM (D) | 1210$_{-35,+40}$ | 0.0 | 10.7% | 2.11 | 0.27 |
| XGBoost (D) | 1190$_{-78,+99}$ | 0.0 | 11.3% | 2.24 | 0.24 |

## E.3 Details on TALENT benchmark results

### E.3.1 Benchmark description

TALENT [36] base contains 300 datasets (120 binary, 80 multiclass, 100 regression). Each dataset is split into 64% training, 16% validation, and 20% test sets.

**Baselines.** We rely on precomputed baselines provided by the authors of the TALENT benchmark [36] (for the TALENT extensions which we use for the large-rows slice and the many-class slice) or the TabICLv2 paper [30] (for the main TALENT slice).

**Metrics.** Following the TALENT paper and [30], we use accuracy for classification and rmse for regression.

**Datasets.** Following [30], we exclude the 26 development datasets used for TabPFN-2 / TabICLv2 development from the main TALENT benchmark.

### E.3.2 Per-task-type breakdown

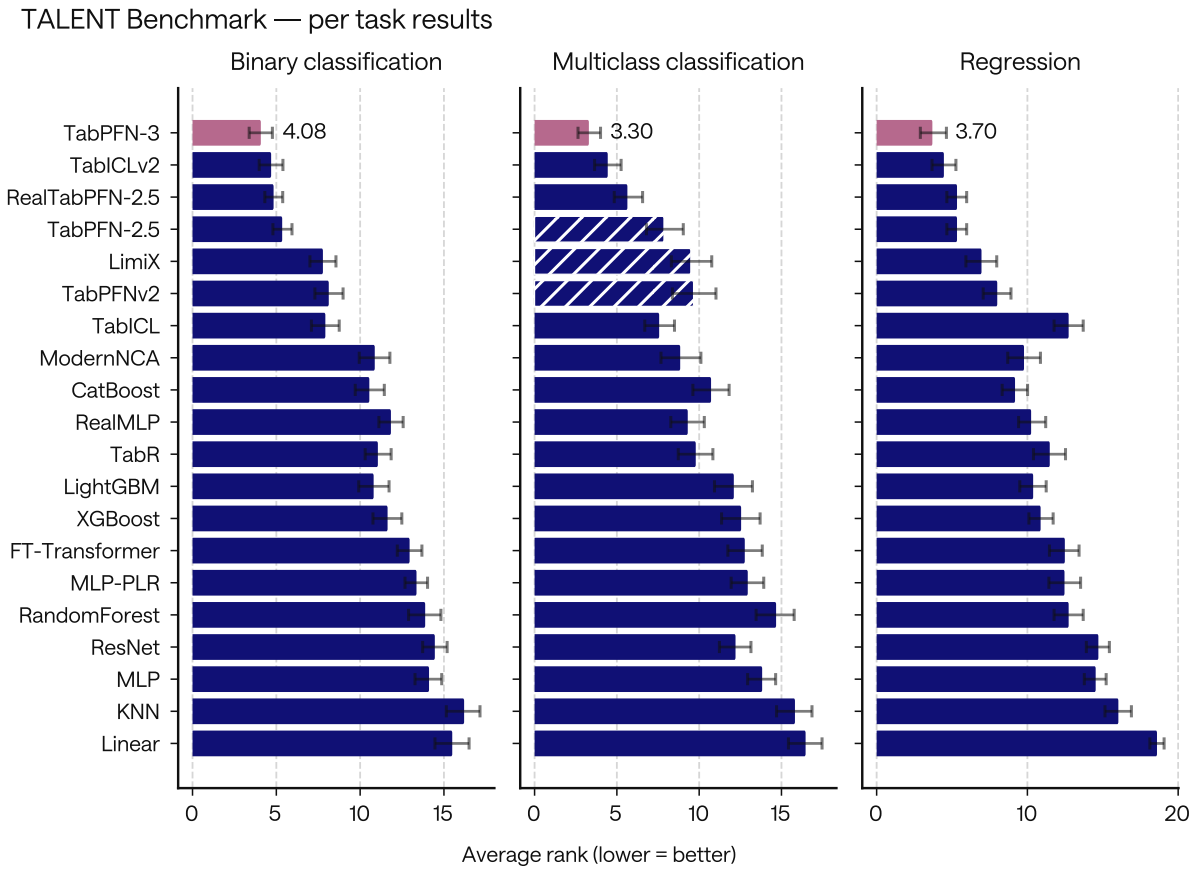



**Figure 28. Average rank on the TALENT benchmark broken down by task type (regression, binary classification, multiclass classification), using the TabICLv2 evaluation protocol from Qu et al. [30].** Bars show mean rank (lower is better); error bars are 95% bootstrap confidence intervals over datasets. Hatched bars mark methods with KNN-imputed scores. TabPFN-2.5, LimiX, and TabPFNv2 share a 10-class cap, so their scores on the 12 multiclass datasets with >10 classes are KNN-imputed.

### E.3.3 Many-class TALENT subset

We report results on the subset of TALENT [36] datasets with more than 50 classes, which yields 4 datasets with 100 classes, including 3 from the same family. While limited in number, these complement the results on synthetic data from Section 3.2.2. Results are shown in Figure 29.

| Dataset | Classes | Samples | Feat. |
|---|---|---|---|
| one-hundred-plants-margin | 100 | 1,600 | 64 |
| one-hundred-plants-shape | 100 | 1,600 | 64 |
| one-hundred-plants-texture | 100 | 1,599 | 64 |
| helena | 100 | 65,196 | 27 |

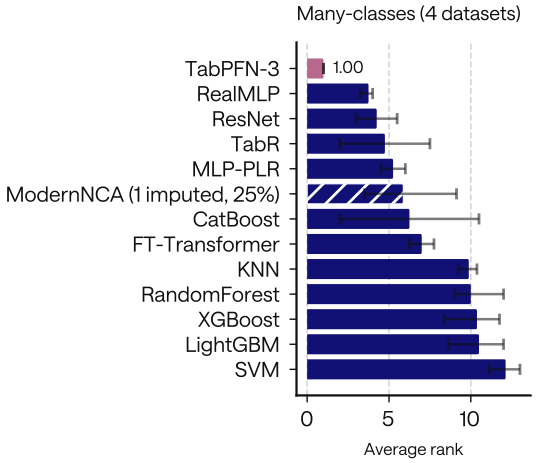



**Figure 29. Average rank on the many-classes TALENT slice (4 datasets, all 100 classes).** Three are the `one-hundred-plants` variants (margin / shape / texture, ≈1.6 k samples each) and one is `helena` (65 k samples).

### E.3.4 Large rows subset

We report here the list of datasets in the large-rows subset of TALENT we use in Section 3.2.1. The datasets are filtered for >100k samples and ≤1M training samples from the TALENT base and large extension. We report the model ranking in Figure 30.

| Dataset | Samples | Feat. | Task |
|---|---|---|---|
| microsoft | 1,200,192 | 136 | Reg. |
| poker-hand | 1,025,009 | 10 | Multi. |
| BNG(credit-a) | 1,000,000 | 15 | Binary |
| Higgs | 1,000,000 | 28 | Binary |
| Smoking_and_Drinking_Dataset_with_body_signal | 991,346 | 23 | Binary |
| yahoo | 709,877 | 699 | Reg. |
| Data_Science_for_Good_Kiva_Crowdfunding | 671,205 | 11 | Multi. |
| covertype | 581,012 | 54 | Multi. |
| CDC_Diabetes_Health_Indicators | 253,680 | 21 | Binary |
| accelerometer | 153,004 | 4 | Multi. |
| walking-activity | 149,332 | 4 | Multi. |
| Rain_in_Australia | 145,460 | 18 | Multi. |
| customer_satisfaction_in_airline | 129,880 | 21 | Binary |
| diabetes_130-us_hospitals | 101,766 | 20 | Binary |

**Table 13.** Datasets in the large-rows TALENT slice.

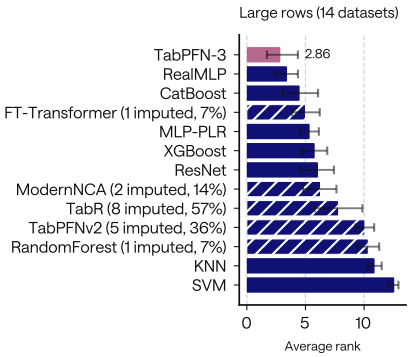



**Figure 30.** Average rank on the large-rows (100k-1M rows) TALENT slice.

### E.3.5 Details

**Per-dataset ranking.** For each (dataset, split) we rank all methods by their score (best = 1; ties get average ranks). The reported *mean rank* of a method is the average of these for ranks across all (dataset, split) pairs in the slice.

**Bootstrap confidence intervals.** 95% confidence intervals are non-parametric bootstrap over datasets: for each of $B = 2{,}000$ replicates we resample the (dataset, split) pairs with replacement and recompute each method's mean rank, then take the empirical 2.5/97.5 percentiles across replicates.

## E.4 Details on TabSTAR Text-Tabular Benchmark results

The TabSTAR benchmark is a union of previous text-tabular benchmarks: the Multimodal AutoML Benchmark [47], Grinsztajn et al. [48], and CARTE [49]. After deduplication and exclusion of unavailable

datasets, the final benchmark contains 50 datasets: 15 classification and 35 regression tasks.[10] Each model is run 5 times, with per-task metrics AUROC (binary classification), log-loss (multiclass), and RMSE (regression); results are normalized with MinMax scaling to the $[0, 1]$ range. As in the original paper [37], we limit each run to up to 100,000 examples. Figure 31 shows the results for classification, for which the TabSTAR model was reportedly the state of the art; we see that the TabPFN API family significantly outperforms it. Figure 32 shows the equivalent regression performance.

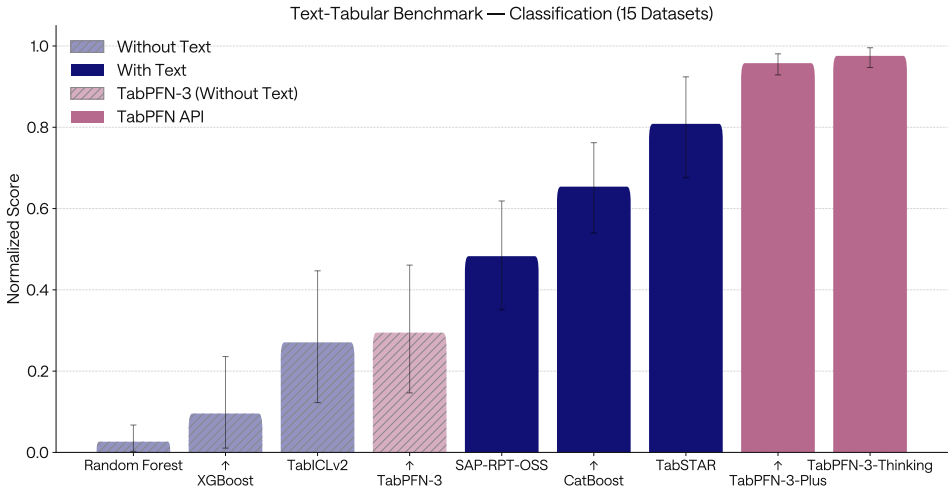



**Figure 31. Performance on the classification tasks of the TabSTAR text-tabular collection.** TabPFN-3-Plus (Thinking) and TabPFN-3-Plus significantly outperform the text-aware TabSTAR, which was otherwise the state-of-the-art reference for this task type.

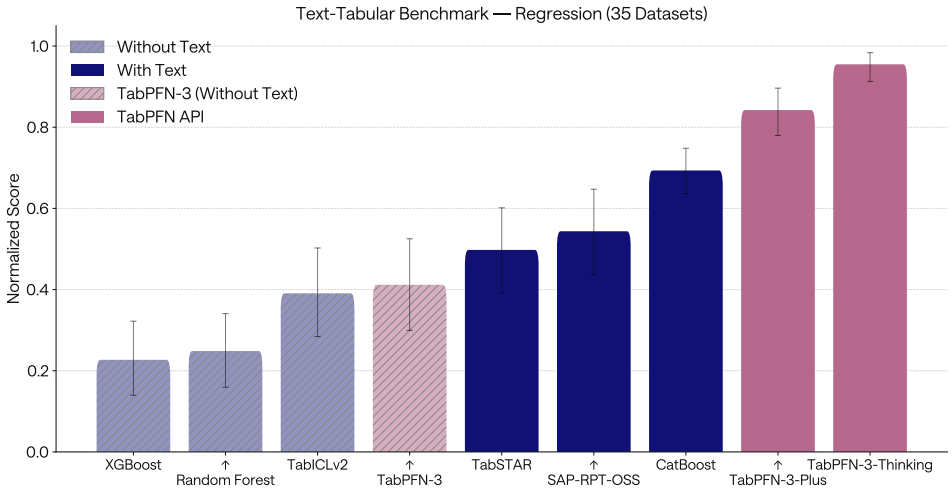



**Figure 32. Performance on the regression tasks of the TabSTAR text-tabular collection.** TabPFN-3-Plus (Thinking) and TabPFN-3-Plus significantly outperform all baselines.

## E.5 Per-dataset results on RelBenchV1

We report per-dataset results for entity regression and entity classification as well as aggregate metrics in Table 14 and Table 15, respectively.

[10]The TabSTAR paper reports 14 classification tasks, having mistakenly treated *Spotify Genres* as a regression dataset.

**Table 14. RelBenchV1 entity classification: per-task ROC AUC×100. Bold** = best per task / column, underlined = second best. Methods marked with * in their name (KumoRFMv1, $\text{RT}_{\text{zero}}$ ) indicate methods that are likely following a different evaluation protocol, which overestimates model performance. KumoRFMv2 was reevaluated by us using the evaluation scripts provided by the authors; cells marked * in this row are imputed with the author's results because they were not supported in the evaluation scripts.

| Method | f1 | | avito | | event | | trial | amazon | | stack | | hm | Avg AUROC ↑ | Rank ↓ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | dnf | top3 | click | visit | repeat | ignore | out | user | item | eng | badge | churn | | |
| RelGNN | 75.29 | 85.69 | 68.23 | 66.18 | **79.61** | 86.18 | 71.24 | **70.99** | 82.64 | **90.75** | **88.98** | **70.93** | **78.06** | **2.83** |
| RelGT | 75.87 | 83.52 | 68.30 | 66.78 | 76.09 | 81.57 | 68.61 | 70.39 | 82.55 | 90.53 | 86.32 | 69.27 | 76.65 | 4.50 |
| GraphSAGE | 72.62 | 75.54 | 65.90 | 66.20 | 76.89 | 81.62 | 68.60 | 70.42 | **82.81** | 90.59 | 88.86 | 69.88 | 75.83 | 5.17 |
| KumoRFMv1* | **82.41** | **91.07** | 64.85 | 64.11 | 76.08 | **89.20** | 70.79 | 67.29 | 79.93 | 87.09 | 80.00 | 67.71 | 76.71 | 6.75 |
| Griffin | 57.70 | 82.50 | 45.90 | 60.70 | 71.88 | 83.27 | 51.00 | 62.30 | 69.00 | 77.50 | 73.50 | 60.20 | 66.29 | 9.92 |
| $\text{RT}^{*}_{\text{zero}}$ | 81.20 | 89.30 | 59.50 | 61.80 | 73.22 | 77.47 | 51.80 | 64.00 | 70.90 | 75.70 | 80.10 | 62.80 | 70.65 | 8.67 |
| RDBLearn | 70.87 | 79.69 | 69.04 | 65.49 | 75.04 | 82.52 | 71.58 | 67.57 | 82.07 | 89.39 | 85.26 | 68.05 | 75.55 | 6.83 |
| RDBLearn + v2.5 | 71.72 | 77.60 | 65.72 | 66.47 | 75.55 | 78.65 | 72.90 | 69.74 | 82.18 | 90.23 | 82.81 | 70.11 | 75.31 | 6.46 |
| RDBLearn + v3 | 71.72 | 82.72 | **69.06** | 66.76 | 76.81 | 73.70 | 72.89 | 69.35 | 82.46 | 90.59 | 85.98 | 70.06 | 76.01 | 4.83 |
| KumoRFMv2 | 72.03 | 82.09 | 67.42* | **69.41*** | 79.34 | 78.86 | 72.03* | 67.71 | 80.18 | 88.69 | 85.40 | 67.81 | 75.91 | 5.75 |
| **TabPFN-REL** | 70.74 | 79.98 | 67.09 | 66.68 | 77.11 | 85.38 | **76.43** | 70.27 | **82.81** | 90.66 | 85.17 | 70.55 | 76.91 | 4.29 |

**Table 15. RelBenchV1 entity regression: per-task MAE. Bold** = best per task / column (ties bolded together at the displayed precision), underlined = second best. Right column: LightGBM-normalised mean $S_{\text{KumoNorm}} = \text{mean}_t \, \text{MAE}_t / \text{MAE}_t^{\text{LightGBM}}$. Methods marked with * in their name (KumoRFMv1, $\text{RT}_{\text{zero}}$ ) indicate methods that are likely following a different evaluation protocol, which overestimates model performance. KumoRFMv2 was reevaluated by us using the evaluation scripts provided by the authors; cells marked * in this row are imputed with the author's results because they were not supported in the evaluation scripts. On one task (`rel-amazon/item`) marked with ** we fall back to the default context size of 5000, because the context exceeded the API's 30MB size limits.

| Method | f1 | avito | event | trial | | amazon | | stack | hm | Avg $S_{\text{KumoNorm}}$ ↓ | Rank ↓ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | pos | ctr | attend | adverse | succ | user | item | votes | sales | | |
| RelGNN | 3.798 | 0.037 | 0.238 | 44.461 | **0.301** | **14.230** | 48.767 | **0.065** | 0.054 | **0.861** | 3.72 |
| RelGT | 3.917 | 0.035 | 0.250 | 43.992 | 0.326 | 14.267 | 48.922 | **0.065** | 0.054 | 0.870 | 4.67 |
| GraphSAGE | 4.022 | 0.041 | 0.258 | 44.473 | 0.400 | 14.313 | 50.053 | **0.065** | 0.056 | 0.918 | 6.39 |
| KumoRFMv1* | **2.747** | 0.035 | 0.264 | 58.231 | 0.417 | 16.161 | 55.254 | **0.065** | **0.040** | 0.908 | 6.06 |
| Griffin | 4.460 | 0.050 | 0.461 | 78.232 | 0.463 | 35.590 | 53.214 | 0.092 | 0.151 | 1.471 | 10.56 |
| $\text{RT}^{*}_{\text{zero}}$ | 2.901 | 0.058 | 0.379 | 73.999 | 0.455 | 18.802 | 57.996 | 0.110 | 0.089 | 1.240 | 9.44 |
| RDBLearn | 3.834 | 0.034 | **0.237** | 43.913 | 0.424 | 14.540 | 48.559 | 0.068 | 0.064 | 0.906 | 5.11 |
| RDBLearn + v2.5 | 3.930 | 0.034 | 0.243 | 43.409 | 0.429 | 14.463 | 49.053 | 0.068 | 0.066 | 0.913 | 6.28 |
| RDBLearn + v3 | 3.835 | 0.034 | 0.245 | 43.290 | 0.375 | 14.720 | 50.097 | 0.068 | 0.064 | 0.898 | 5.89 |
| KumoRFMv2 | 4.022 | 0.033 | 0.241 | 41.974 | 0.433* | 14.627 | **45.352** | **0.065**** | 0.043 | 0.866 | 4.33 |
| **TabPFN-REL** | 3.757 | **0.031** | 0.241 | **40.202** | 0.385 | 14.359 | 46.199 | 0.068 | 0.059 | 0.864 | **3.56** |

# F Additional Details on Internal Benchmarks

## F.1 Methodology

**Metric Normalization.** To aggregate heterogeneous metrics across datasets, we apply a per-fold min–max normalization. For each (dataset, fold) pair and metric $m$, we rescale a model's raw score $s_m^{(b)}$ as

$$\tilde{s}_m^{(b)} = \frac{s_m^{(b)} - \min_{b' \in \mathcal{B}} s_m^{(b')}}{\max_{b' \in \mathcal{B}} s_m^{(b')} - \min_{b' \in \mathcal{B}} s_m^{(b')}}, \tag{1}$$

where $\mathcal{B}$ denotes the set of models we evaluate. This allows the model scores to live on a comparable $[0, 1]$ scale for each (dataset, fold) combination. We treat the tuned and default versions of a model as two different models. For lower-is-better metrics (e.g. RMSE, cross-entropy loss), we apply the additional transformation $\tilde{s}_m \mapsto 1 - \tilde{s}_m$, so that all metrics are higher-is-better on a common scale and can be meaningfully averaged or ranked across datasets and metric types.

**Statistical significance.** To assess whether performance differences between models are statistically significant, we report critical difference (CD) diagrams using `scikit-posthocs` [76]. The critical difference diagram from `scikit-posthocs` summarizes the statistical comparison of methods across multiple datasets. Average ranks are computed per method across all datasets, with lower ranks indicating better performance. Methods connected by a horizontal bar are not significantly different from each other. To assess statistical significance, we use a Friedman test followed by a Conover post hoc analysis at the significance level $\alpha = 0.05$.

## F.2 Large Data Benchmark Details

Classification datasets span domains including healthcare (patient survival, disease diagnosis), customer analytics (satisfaction, credit risk), insurance (claim prediction), microfinance (loan outcomes), and high-energy physics (signal/background classification). Regression datasets cover retail sales forecasting, climate and weather modeling, food delivery logistics, and e-commerce price prediction. All 4 regression datasets use temporal train/test splits reflecting real-world deployment conditions where the test period strictly follows the training period. For classification, all datasets are IID. These datasets are selected to have between 100K and 1M training rows, and fewer than 200 features, which is the regime TabPFN-3 was designed for.

Figures 33 and 34 show critical difference diagrams for ROC-AUC and RMSE respectively, based on average ranks across all datasets in each benchmark.

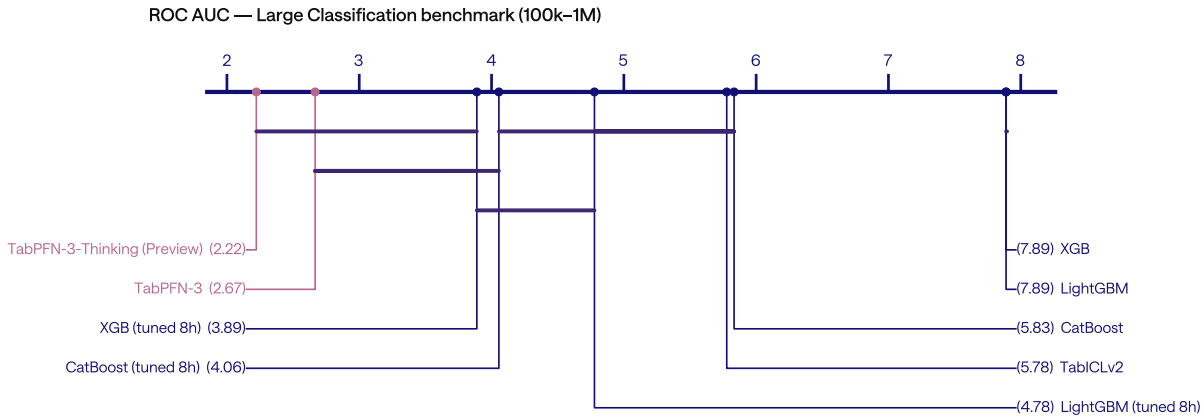



**Figure 33. Critical difference diagram for ROC-AUC on the large-scale classification benchmark (100k–1M training rows).** TabPFN-3 ranks first (avg. rank 2.11). Its rank differences to the 8-hour-tuned XGB/CatBoost baselines are not statistically significant, while it ranks significantly ahead of tuned LightGBM, all default GBTs, and TabICLv2. Bars connect methods whose rank differences are not statistically significant at $\alpha = 0.05$ under a Conover-Friedman post-hoc test [76]..

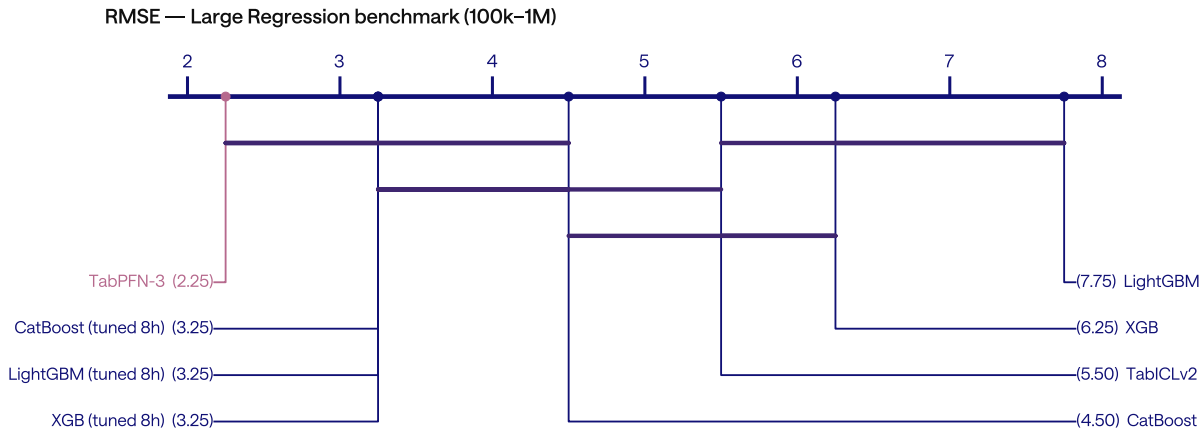



**Figure 34. Critical difference diagram for RMSE on the large-scale regression benchmark (100k–1M rows, 4 datasets, temporal splits).** Methods are ranked per (dataset, split); lower rank is better. TabPFN-3 achieves the best average rank (2.25). Its rank differences to the three 8h-tuned GBDTs and untuned CatBoost are not statistically significant, while it ranks significantly ahead of the remaining methods. Bars connect methods whose rank differences are not statistically significant at $\alpha = 0.05$ under a Conover-Friedman post-hoc test [76].

## F.3 Synthetic Many-Class Benchmark Construction

Continuous regression targets are partitioned into $K = 100$ bins using quantile-based bin edges whose spacings are drawn from a Dirichlet($\alpha$=5.0) distribution, producing realistic class imbalance. Bins with fewer than 10 samples are merged with their nearest neighbour to guarantee sufficient representation for inner cross-validations. Class labels are then randomly permuted to remove the implicit ordinal structure inherited from the regression target.

tasets from TabArena whose targets have heavy point masses or too few distinct values to fill 100 quantile bins meaningfully — wine_quality (7 unique values), Food_Delivery_Time (45, discrete times), Fiat-500 (222, discrete prices), and QSAR-TID-11 (concentrated point masses). Dataset statistics are reported in Table 16. The resulting benchmark retains a large number of classes for most datasets (median $K = 95$), while inducing moderate class imbalance (median IR $= 9.9\times$) without collapsing the label distribution onto a few dominant classes (median $H/\log K = 0.98$).

**Table 16. Synthetic many-class benchmark datasets derived from continuous regression targets.** $N$ is the number of samples before binning. Targets are first partitioned into 100 quantile-based bins with randomized Dirichlet-spaced bin widths, after which bins with fewer than 10 samples are merged into their nearest neighbour. $K$ is the resulting number of classes and **Merged** equals $100 - K$. **Min** and **Max** are the smallest and largest class sizes after merging, and **IR** is their ratio. $H/\log K$ is the Shannon entropy of the class distribution normalized by $\log K$, with 1 corresponding to perfectly balanced classes.

| Dataset | OpenML task | OpenML did | $N$ | $K$ | Merged | Min | Max | IR | $H/\log K$ |
|---|---|---|---|---|---|---|---|---|---|
| airfoil_self_noise | 363612 | 46904 | 1,503 | 80 | 20 | 10 | 42 | 4.2× | 0.984 |
| concrete_compressive_strength | 363625 | 46917 | 1,030 | 60 | 40 | 10 | 28 | 2.8× | 0.991 |
| diamonds | 363631 | 46923 | 53,940 | 100 | 0 | 127 | 1,252 | 9.9× | 0.979 |
| healthcare_insurance_expenses | 363675 | 46931 | 1,338 | 73 | 27 | 10 | 32 | 3.2× | 0.988 |
| houses | 363678 | 46934 | 20,640 | 97 | 3 | 62 | 965 | 15.6× | 0.974 |
| miami_housing | 363686 | 46942 | 13,776 | 95 | 5 | 19 | 339 | 17.8× | 0.970 |
| physiochemical_protein | 363693 | 46949 | 45,730 | 100 | 0 | 108 | 1,214 | 11.2× | 0.982 |
| QSAR_fish_toxicity | 363698 | 46954 | 907 | 51 | 49 | 10 | 37 | 3.7× | 0.980 |
| superconductivity | 363705 | 46961 | 21,263 | 100 | 0 | 28 | 508 | 18.1× | 0.979 |
| Aggregate (mean / median) | — | — | — | 84 / 95 | — | — | — | 9.6× / 9.9× | 0.98 / 0.98 |

## F.4 Quantile Regression: Critical Difference Diagram

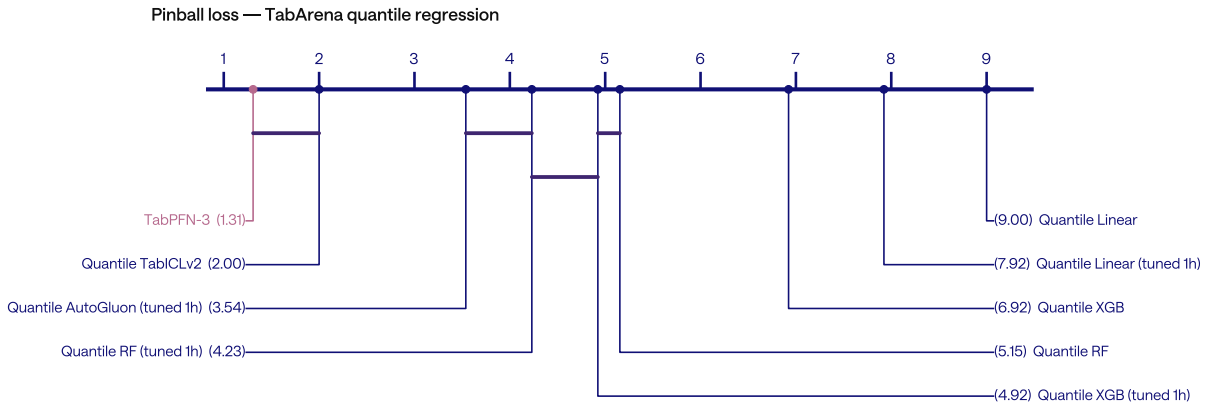



**Figure 35. Critical difference diagram for pinball loss on our quantile regression benchmark.** The quantile regression benchmark is constructed from TabArena regression datasets and evaluated across 10 quantile levels $q \in \{0.1, 0.2, \ldots, 0.9\}$. TabPFN-3 ranks first. Its rank difference to Quantile TabICLv2 is not statistically significant, while it ranks significantly ahead of all remaining baselines. Bars connect methods whose rank differences are not statistically significant at $\alpha = 0.05$ under a Conover-Friedman post-hoc test [76].

## F.5 Synthetic Many Class: Critical Difference Diagram

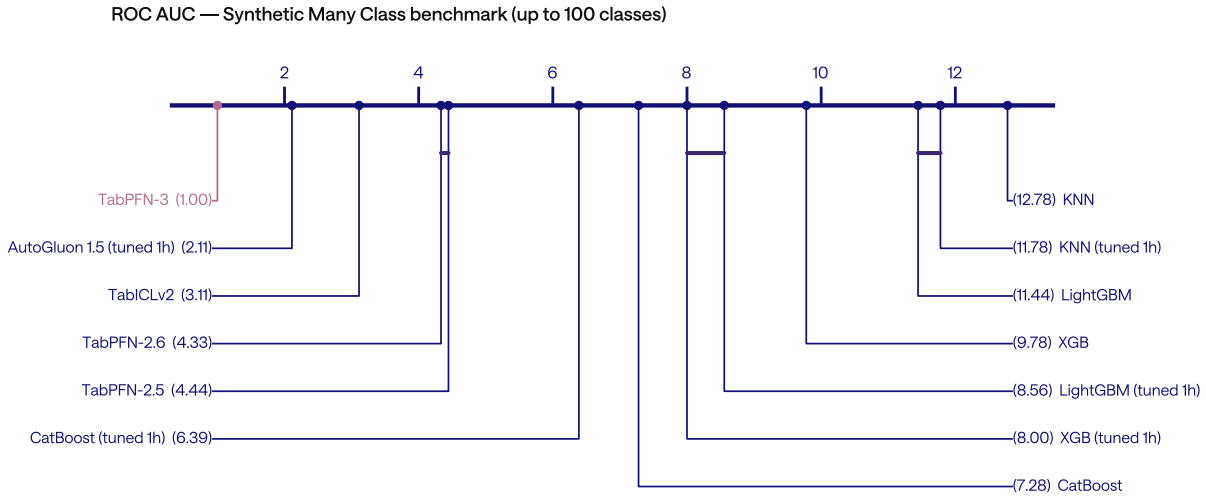
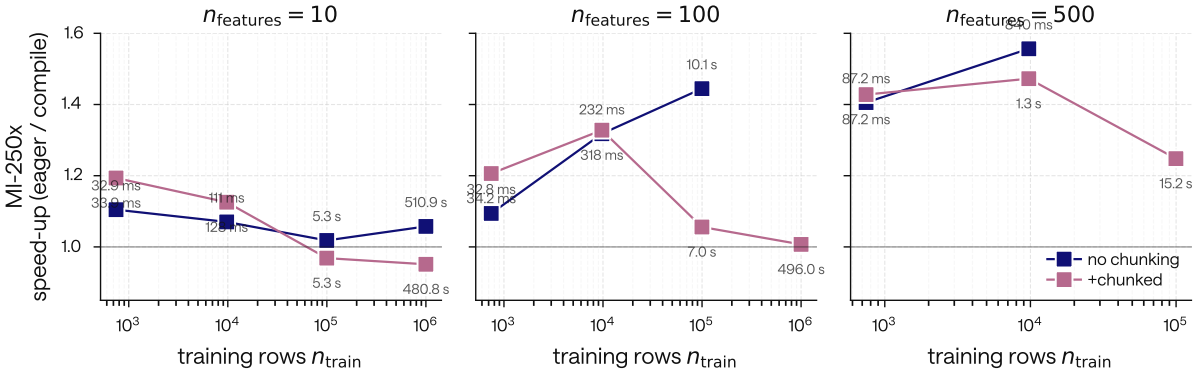



**Figure 36. Critical difference diagram for ROC AUC on the synthetic many-class benchmark (up to 100 classes).** TabPFN-3 is top-ranked on every (dataset, split) pair and ranks significantly ahead of all baselines. Bars connect methods whose rank differences are not statistically significant at $\alpha = 0.05$ under a Conover-Friedman post-hoc test [76].

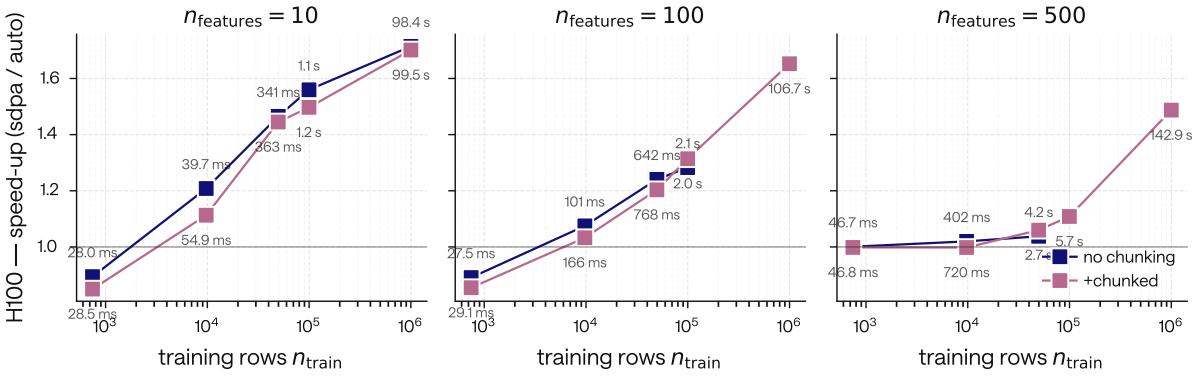
# G Supplementary Inference Time Details

## G.1 Compilation and FlashAttention-3

TabPFN-3 is shipped with two opt-in performance features that target different bottlenecks: `torch.compile` and FlashAttention-3. At the shapes relevant to large-data inference, the bulk of forward-pass cost is dispatch overhead and attention compute, and the speed-ups of `torch.compile` and FlashAttention-3 compose cleanly with our chunking strategy without changing the model's behaviour.




**(a)** MI-250x – speed-up of `torch.compile` over eager of TabPFN-3 forward pass, for $n_{\text{features}} \in \{10, 100, 500\}$. Values above 1 indicate compile wins.




**(b)** H100 – speed-up of the auto backend (Flash Attention 3 where eligible, SDPA fallback elsewhere) over the SDPA-only backend on the TabPFN-3 architecture forward pass, for $n_{\text{features}} \in \{10, 100, 500\}$. Values above 1 indicate FA3 wins.

**Figure 37. Forward-pass inference speed-ups on the TabPFN-3 architecture.** Top: `torch.compile` versus eager execution on MI-250x. Bottom: auto attention backend (Flash Attention 3 where eligible) versus SDPA-only on H100. Marker annotations report absolute execution times.

**torch.compile.** Three hot-path methods are wrapped with `@torch.compile(dynamic=True)`: feature preprocessing plus embedding grouping, the column-chunk processing block (used in the non-row-chunked path), and the row-chunk processing block (used when chunking is enabled). The `dynamic=True` mode keeps a single compiled graph across batch and feature-count variation, so the same compiled artefact serves the whole inference grid without re-tracing.

Figure 37a shows the wall-clock impact on MI-250x. The y-axis is $T_{\text{eager}}/T_{\text{compile}}$, so a value above 1 means compile is faster on that shape; each marker is annotated with the absolute time. `torch.compile` fuses Python-level dispatch into single kernel calls, so it helps most where dispatch is the bottleneck: as $n_{\text{features}}$ grows, more tensor work becomes compile-able per call. In the non-chunked series the speed-up climbs from 1.04–1.15× at $n_{\text{features}} = 10$ to 1.10–1.46× at $n_{\text{features}} = 100$ and 1.40–1.58× at $n_{\text{features}} = 500$. The chunked series shows the same direction with a different shape: chunking already amortises some dispatch overhead by batching the inner loop, so compile's marginal benefit is largest at small $n_{\text{train}}$ (1.21–1.43× at $n_{\text{train}} = 10^3$) and large $n_{\text{features}}$ and converges toward parity (0.95–1.06×) at $n_{\text{train}} \geq 10^5$ for the smaller feature counts, where the residual cost is dominated by attention itself and compile has no further headroom to claim.

**FlashAttention-3.** FlashAttention-3 (FA3) [33] is a Hopper-specific attention kernel that delivers higher throughput and lower memory use than the generic Scaled Dot-Product Attention (SDPA) path. Attention dominates the forward-pass cost of large-$n_{\text{train}}$ inference, so even a constant-factor improvement in the attention kernel translates into a meaningful end-to-end speed-up. We therefore expose FA3 as an auto-detecting backend: on Hopper-class GPUs with the FA3 library installed, the in-context-learning self-attention – which carries the bulk of the attention cost at large $n_{\text{train}}$ – is routed through FA3, while attention sites whose head dimensions are not FA3-eligible silently fall back to SDPA. On non-Hopper

devices (consumer Ada, AMD MI-250x, Blackwell) the same dispatcher selects SDPA.

Figure 37b shows the H100 SDPA-versus-auto comparison in ratio form. The y-axis is $T_{\text{sdpa}}/T_{\text{auto}}$, so a value above 1 means FA3 is faster than SDPA on that shape; each marker is annotated with the absolute auto time so the magnitude being sped up is recoverable. The pattern matches the FA3 design profile. At small training sets ($n_{\text{train}} \leq 1000$) the FA3 dispatch and kernel-launch overhead exceeds the per-call attention work, and SDPA is 10–15% faster ($T_{\text{sdpa}}/T_{\text{auto}} \approx 0.84$–$0.91$ across feature counts). The cross-over arrives sooner the smaller $n_{\text{features}}$: by $n_{\text{train}} = 10^4$ FA3 wins at $n_{\text{features}} = 10$ (1.21×), is roughly even at $n_{\text{features}} = 100$ (1.07×), and at parity at $n_{\text{features}} = 500$ (1.02×). At the inference shapes we care about ($n_{\text{train}} \geq 10^5$) FA3 is the clear win across all feature counts, with the speed-up climbing to 1.49–1.73× at $n_{\text{train}} = 10^6$. Chunking does not interact with the FA3-versus-SDPA comparison: the chunked and non-chunked curves overlap to within run-to-run noise, since chunking changes the outer dispatch loop but leaves the underlying attention-kernel selection intact.

## G.2 Interpretability: SHAP-Value Computation

TabPFN-3's improved, smaller KV cache (Section 2.4.2) can speed up the computation of SHAP values by multiple order of magnitudes. This is because imputation-based approaches to SHAP-value computation reuse the same `fit` on many different forward passes. Figure 38 shows the efficiency gains users can expect from enabling the KV cache during SHAP-value-computation.

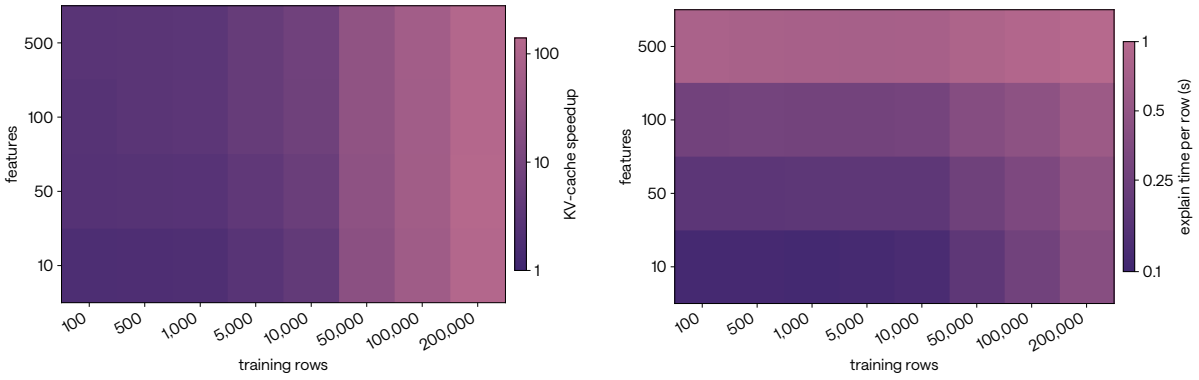



**Figure 38. Efficiency gains for SHAP-value computation with KV-cache across training table dimensions.** All experiments were conducted on a single RTX Pro 6000 Blackwell with a fixed budget of 1024 coalitions and are averaged over 10 repetitions. Left: expected speed-up from using KV cache. Right: expected runtime for computing SHAP values for one test row with KV cache enabled.

# H Detailed Time-Series Forecasting Results on fev-bench

This appendix complements the body Time-Series subsection (Table 1, Figure 20) with the full leaderboards (Table 17), pairwise comparisons (Figure 45), additional qualitative forecasts (Section H and per-task SQL results (Section H).

## Full leaderboards (SQL and MASE)

**Table 17.** Full marginal forecasting performance on fev-bench (100 tasks), all 19 baselines, sorted by skill score. The body table (Table 1) shows the foundation-model + Stat. Ensemble + Seasonal Naive subset. † TabICL-v2 results were produced using the `tabicl[forecast]` package (v2.0.3) on fev-bench v0.7.0; results for this model are not currently available in the official fev-bench results repository.

**(a) SQL (probabilistic)**

| Model | Win (%) | Skill (%) | Runtime (s) | Leak. (%) | # fails |
|---|---|---|---|---|---|
| Chronos-2 | 91.7 | 47.3 | 0.8 | 0 | 0 |
| TabPFN-TS-3 | 73.6 | 43.1 | 234.6 | 0 | 0 |
| TiRex | 83.4 | 42.6 | 0.2 | 1 | 0 |
| TimesFM-2.5 | 78.6 | 42.2 | 1.9 | 10 | 0 |
| Toto-1.0 | 71.6 | 40.7 | 22.1 | 8 | 0 |
| TabPFN-v2-TS | 64.1 | 39.6 | 88.9 | 0 | 2 |
| Moirai-2.0 | 66.2 | 39.3 | 0.3 | 28 | 0 |
| Chronos-Bolt | 66.2 | 38.9 | 0.2 | 0 | 0 |
| Sundial-Base | 47.1 | 33.4 | 8.0 | 1 | 0 |
| TabICL-v2† | 53.8 | 30.8 | 64.7 | 0 | 0 |
| CatBoost (Recursive) | 35.7 | 23.0 | 0.3 | 0 | 0 |
| LightGBM (Recursive) | 33.4 | 21.7 | 0.3 | 0 | 0 |
| AutoARIMA | 39.6 | 20.6 | 19.5 | 0 | 10 |
| Stat. Ensemble | 43.8 | 20.2 | 148.6 | 0 | 11 |
| AutoTheta | 27.1 | 5.5 | 3.3 | 0 | 0 |
| Seasonal Naive | 19.1 | 0.0 | 0.5 | 0 | 0 |
| AutoETS | 32.7 | -26.8 | 3.5 | 0 | 3 |
| Naive | 12.6 | -45.4 | 0.5 | 0 | 0 |
| Drift | 9.7 | -45.8 | 0.5 | 0 | 0 |

**(b) MASE (point)**

| Model | Win (%) | Skill (%) | Runtime (s) | Leak. (%) | # fails |
|---|---|---|---|---|---|
| Chronos-2 | 86.9 | 35.5 | 0.8 | 0 | 0 |
| TabPFN-TS-3 | 69.8 | 30.6 | 234.6 | 0 | 0 |
| TimesFM-2.5 | 74.9 | 30.2 | 1.9 | 10 | 0 |
| TiRex | 76.9 | 30.0 | 0.2 | 1 | 0 |
| Toto-1.0 | 66.3 | 28.2 | 22.1 | 8 | 0 |
| TabPFN-v2-TS | 58.5 | 27.6 | 88.9 | 0 | 2 |
| Moirai-2.0 | 61.4 | 27.3 | 0.3 | 28 | 0 |
| Chronos-Bolt | 60.7 | 26.5 | 0.2 | 0 | 0 |
| Sundial-Base | 53.4 | 24.7 | 8.0 | 1 | 0 |
| CatBoost (Recursive) | 54.0 | 23.7 | 0.3 | 0 | 0 |
| LightGBM (Recursive) | 50.3 | 22.4 | 0.3 | 0 | 0 |
| Stat. Ensemble | 46.7 | 15.7 | 148.6 | 0 | 11 |
| AutoARIMA | 36.0 | 11.2 | 19.5 | 0 | 10 |
| AutoTheta | 34.2 | 11.0 | 3.3 | 0 | 0 |
| TabICL-v2† | 33.2 | 7.0 | 64.7 | 0 | 0 |
| AutoETS | 33.5 | 2.3 | 3.5 | 0 | 3 |
| Seasonal Naive | 20.0 | 0.0 | 0.5 | 0 | 0 |
| Naive | 18.0 | -16.7 | 0.5 | 0 | 0 |
| Drift | 15.3 | -18.1 | 0.5 | 0 | 0 |

## Qualitative forecast examples

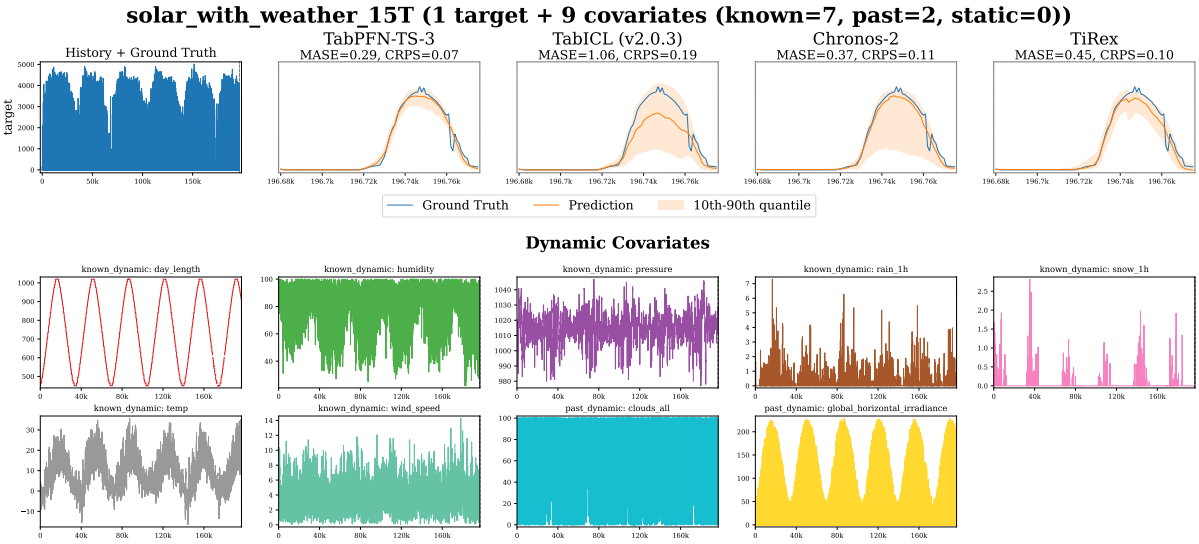



**Figure 39.** `solar_with_weather_15T` — 15-minute solar generation with weather covariates.

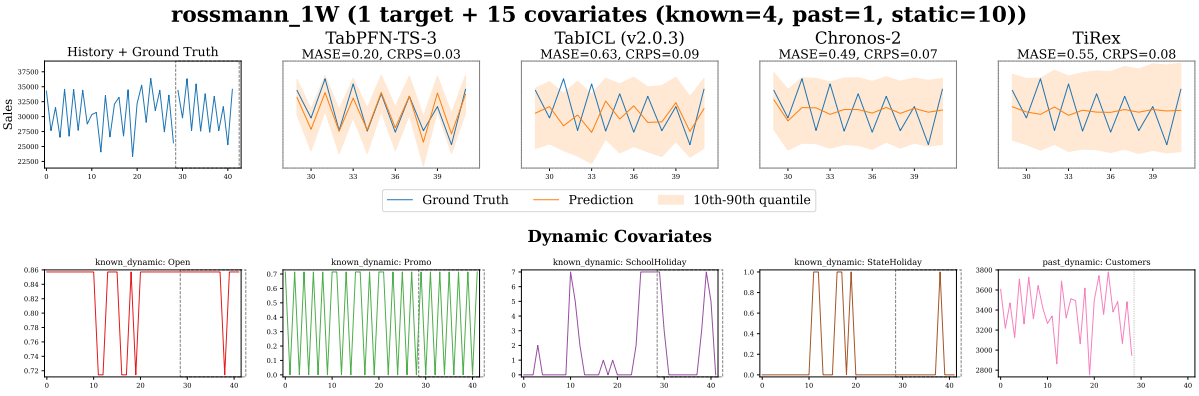



**Figure 40.** `rossmann_1W` — weekly Rossmann store sales (series 1).

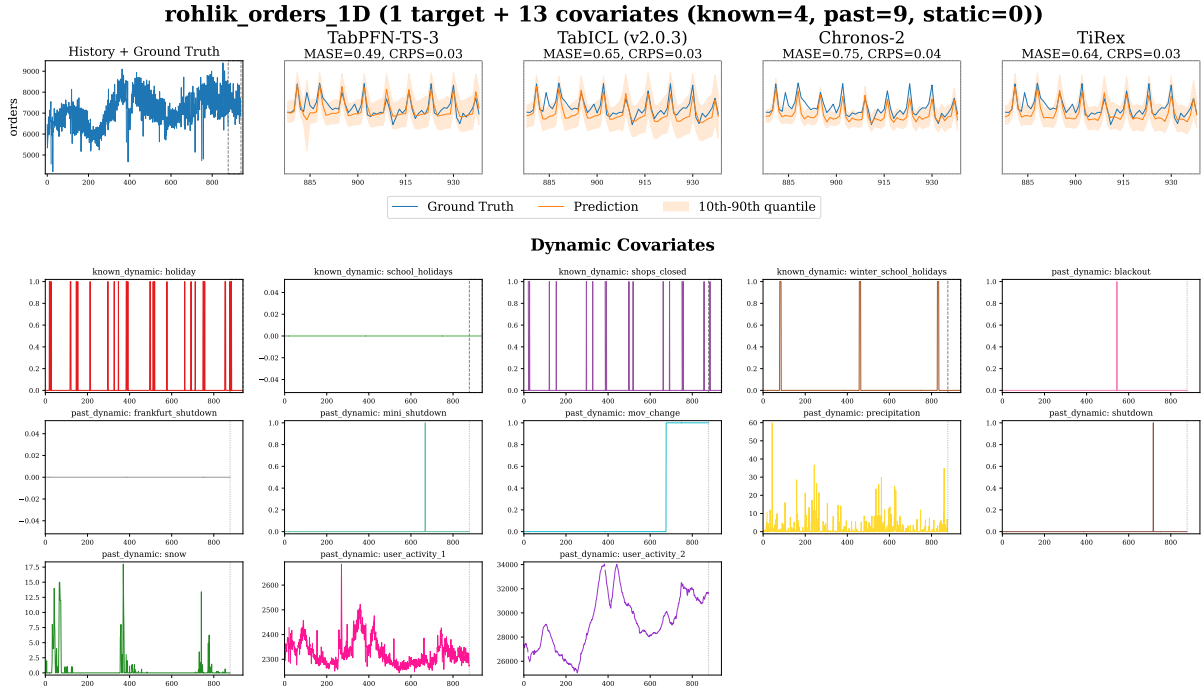



**Figure 41.** `rohlik_orders_1D` — daily online-grocery orders.

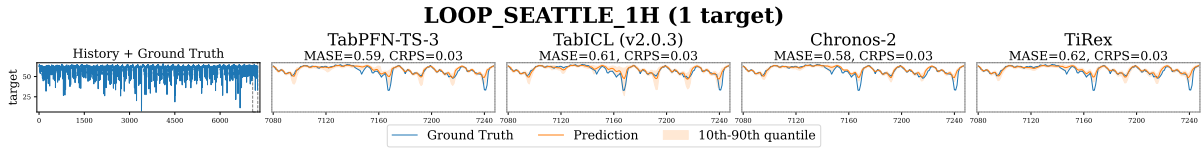



**Figure 42.** `LOOP_SEATTLE_1H` — hourly Seattle freeway loop-detector counts.

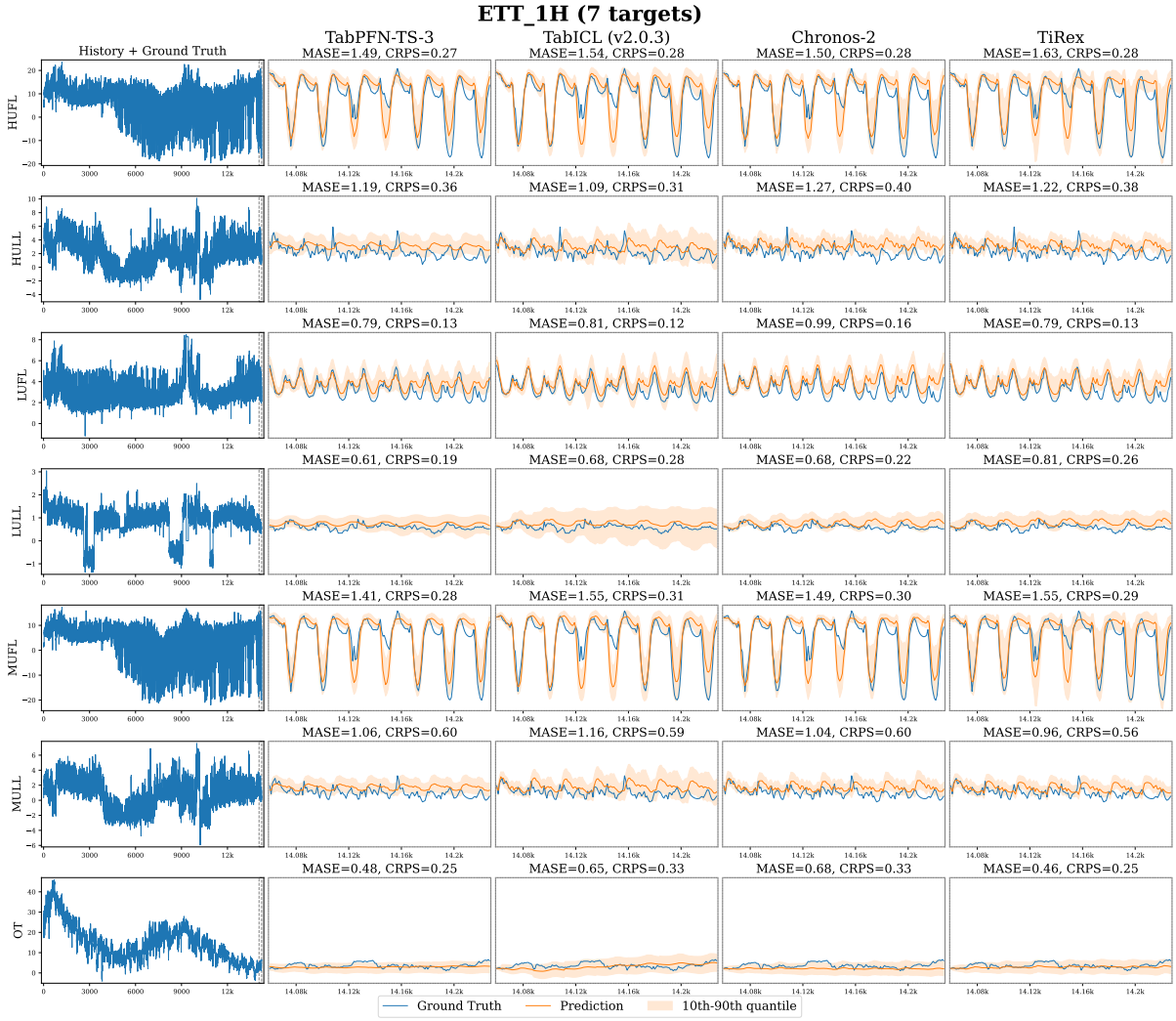



**Figure 43.** `ETT_1H` — hourly Electricity Transformer Temperature.

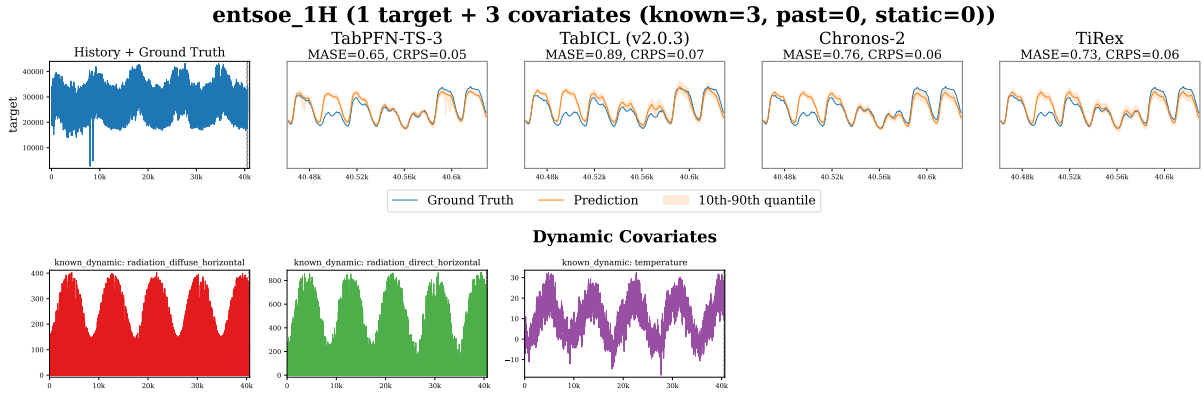



**Figure 44.** `entsoe_1H` — hourly ENTSO-E European electricity load.

## Pairwise skill-score heatmaps

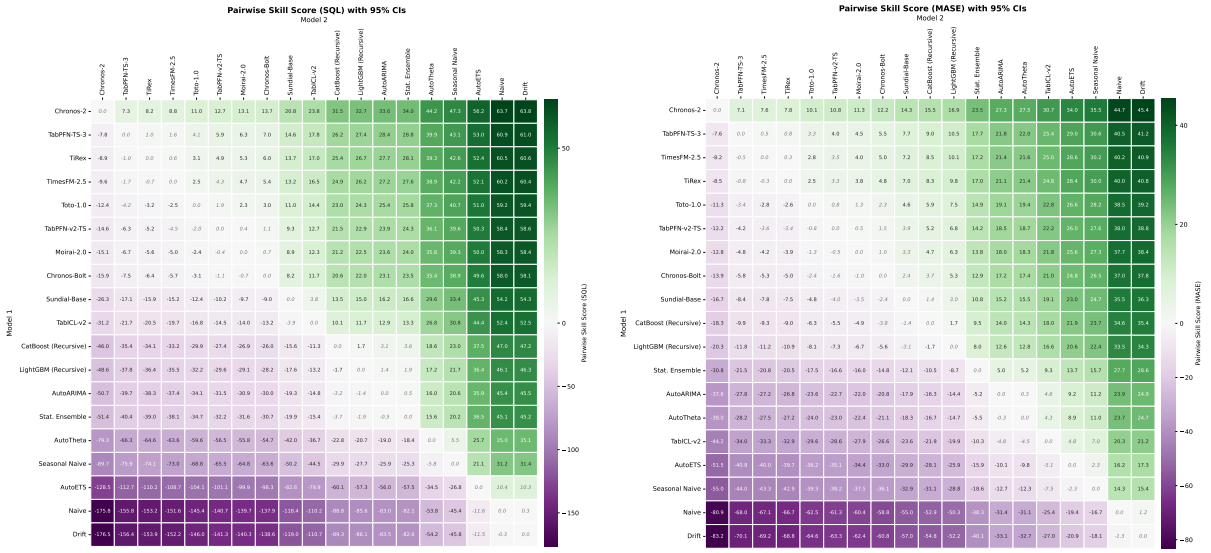



**Figure 45.** Pairwise skill-score comparison on fev-bench (100 tasks) under SQL (left) and MASE (right). Cell $(i, j)$ is the skill score of model $i$ relative to model $j$, with 95% confidence intervals from bootstrapped resampling; cells whose interval overlaps zero are shown in italics. Rows and columns are ordered by overall skill score. Best viewed on screen.

## fev-bench per-task SQL leaderboard

**Table 18. Per-task SQL on fev-bench (100 tasks).** Lower is better; values are after leakage and failure imputation. Per-row top-three are highlighted with gold / silver / bronze backgrounds. Columns are the ten models with the most medal placements; ordered by overall SQL skill score. Values exceeding $10^3$ are capped for layout.

| Task name | Chronos-2 | TabPFN-TS-3 | TiRex | TimesFM-2.5 | Toto-1.0 | TabPFN-v2-TS | Moirai-2.0 | Chronos-Bolt | StatEns | AutoETS |
|---|---|---|---|---|---|---|---|---|---|---|
| ETT_15T | 0.546 | 0.626 | 0.568 | 0.577 | 0.593 | 0.602 | 0.574 | 0.574 | 0.762 | 1.263 |
| ETT_1D | 1.132 | 1.138 | 1.101 | 1.144 | 1.143 | 1.230 | 1.132 | 1.132 | 1.271 | 1.356 |
| ETT_1H | 0.883 | 0.908 | 0.874 | 0.882 | 0.873 | 0.933 | 0.944 | 0.944 | 1.272 | 1.765 |
| ETT_1W | 2.320 | 2.252 | 2.265 | 2.249 | 2.281 | 2.411 | 2.280 | 2.280 | 2.407 | 2.394 |
| LOOP_SEATTLE_1D | 0.779 | 0.769 | 0.792 | 0.774 | 0.831 | 0.780 | 0.805 | 0.805 | 0.820 | 0.825 |
| LOOP_SEATTLE_1H | 0.639 | 0.667 | 0.656 | 0.621 | 0.698 | 0.679 | 0.765 | 0.765 | 1.501 | 2.639 |
| LOOP_SEATTLE_5T | 0.533 | 0.710 | 0.549 | 0.595 | 0.561 | 0.641 | 0.710 | 0.710 | 1.044 | 1.155 |
| M_DENSE_1D | 0.646 | 0.757 | 0.746 | 0.708 | 0.842 | 0.756 | 0.759 | 0.759 | 0.965 | 1.073 |
| M_DENSE_1H | 0.585 | 0.585 | 0.587 | 0.556 | 0.621 | 0.646 | 0.595 | 0.595 | 1.127 | 59.020 |
| SZ_TAXI_15T | 0.393 | 0.399 | 0.396 | 0.397 | 0.401 | 0.429 | 0.413 | 0.413 | 0.560 | 2.355 |
| SZ_TAXI_1H | 0.398 | 0.407 | 0.405 | 0.416 | 0.418 | 0.494 | 0.426 | 0.426 | 0.689 | $>10^3$ |
| aust...tourism | 0.677 | 0.695 | 0.786 | 0.732 | 0.890 | 0.699 | 0.918 | 0.928 | 0.730 | 0.762 |
| bizitobs_l2c_1H | 0.301 | 0.374 | 0.366 | 0.326 | 0.370 | 0.354 | 0.342 | 0.342 | 0.634 | 0.718 |
| bizitobs_l2c_5T | 0.411 | 0.370 | 0.679 | 0.461 | 0.595 | 0.485 | 0.757 | 0.757 | 0.720 | 0.731 |
| boomlet_1062 | 0.552 | 0.554 | 0.555 | 0.573 | 0.548 | 0.708 | 0.593 | 0.639 | 0.985 | 1.309 |
| boomlet_1209 | 0.680 | 0.768 | 0.729 | 0.705 | 0.645 | 1.016 | 0.756 | 0.784 | 2.469 | 1.264 |
| boomlet_1225 | 0.186 | 0.199 | 0.188 | 0.190 | 0.183 | 0.215 | 0.195 | 0.203 | 0.280 | 0.318 |
| boomlet_1230 | 1.201 | 1.292 | 1.186 | 1.187 | 1.138 | 1.613 | 1.286 | 1.266 | 3.390 | $>10^3$ |
| boomlet_1282 | 0.421 | 0.413 | 0.409 | 0.403 | 0.407 | 0.425 | 0.427 | 0.462 | 0.739 | 0.914 |
| boomlet_1487 | 0.423 | 0.447 | 0.427 | 0.412 | 0.400 | 0.745 | 0.456 | 0.482 | 0.681 | 0.724 |
| boomlet_1631 | 0.572 | 0.622 | 0.598 | 0.579 | 0.581 | 0.697 | 0.591 | 0.619 | 0.851 | 0.721 |
| boomlet_1676 | 0.569 | 0.602 | 0.571 | 0.563 | 0.554 | 0.831 | 0.573 | 0.608 | 0.850 | 0.756 |
| boomlet_1855 | 0.462 | 0.504 | 0.450 | 0.473 | 0.452 | 0.623 | 0.465 | 0.470 | 1.123 | 1.185 |
| boomlet_1975 | 0.133 | 0.251 | 0.192 | 0.167 | 0.126 | 0.207 | 0.220 | 0.179 | 0.548 | 0.611 |
| boomlet_2187 | 0.712 | 0.835 | 0.711 | 0.802 | 0.764 | 0.934 | 0.807 | 0.775 | 1.273 | 1.307 |
| boomlet_285 | 0.290 | 0.354 | 0.345 | 0.397 | 0.319 | 0.713 | 0.427 | 0.477 | 1.262 | 1.203 |
| boomlet_619 | 0.323 | 0.326 | 0.341 | 0.340 | 0.310 | 0.331 | 0.329 | 0.471 | 0.777 | 0.894 |
| boomlet_772 | 0.283 | 0.305 | 0.296 | 0.295 | 0.281 | 0.330 | 0.314 | 0.339 | 1.179 | $>10^3$ |
| boomlet_963 | 0.717 | 0.786 | 0.718 | 0.739 | 0.720 | 0.796 | 0.751 | 0.779 | 1.335 | 1.609 |

| Task name | Chronos-2 | TabPFN-TS-3 | TiRex | TimesFM-2.5 | Toto-1.0 | TabPFN-v2-TS | Moirai-2.0 | Chronos-Bolt | StatEns | AutoETS |
|---|---|---|---|---|---|---|---|---|---|---|
| ecdc_ili | 2.271 | 2.457 | 2.411 | 2.215 | 2.554 | 2.382 | 2.454 | 2.653 | 3.837 | 4.079 |
| entsoe_15T | 0.454 | 0.648 | 0.469 | 0.471 | 0.591 | 0.484 | 0.478 | 0.506 | 0.781 | 3.029 |
| entsoe_1H | 0.429 | 0.385 | 0.470 | 0.468 | 0.480 | 0.442 | 0.487 | 0.457 | 0.892 | 1.905 |
| entsoe_30T | 0.434 | 0.579 | 0.523 | 0.566 | 0.496 | 0.512 | 0.488 | 0.529 | 0.847 | 2.493 |
| epf_be | 0.503 | 0.533 | 0.527 | 0.494 | 0.565 | 0.532 | 0.528 | 0.573 | 1.213 | 1.534 |
| epf_de | 0.491 | 0.437 | 1.032 | 1.030 | 1.106 | 0.440 | 1.016 | 1.021 | 1.167 | 1.401 |
| epf_fr | 0.362 | 0.374 | 0.401 | 0.409 | 0.426 | 0.331 | 0.409 | 0.439 | 1.146 | 0.899 |
| epf_np | 0.658 | 0.633 | 0.966 | 1.171 | 1.037 | 0.659 | 0.925 | 0.971 | 1.284 | 1.933 |
| epf_pjm | 0.382 | 0.382 | 0.404 | 0.426 | 0.452 | 0.427 | 0.441 | 0.422 | 0.487 | 0.914 |
| ercot_1D | 0.869 | 0.845 | 0.818 | 0.830 | 0.880 | 0.981 | 0.947 | 0.916 | 1.255 | 1.382 |
| ercot_1H | 1.029 | 1.108 | 1.065 | 1.151 | 1.095 | 1.208 | 1.098 | 1.138 | 1.260 | 2.676 |
| ercot_1M | 0.755 | 0.755 | 0.806 | 0.772 | 1.007 | 0.903 | 0.973 | 0.773 | 0.762 | 0.756 |
| ercot_1W | 0.966 | 0.996 | 0.955 | 0.932 | 1.060 | 1.228 | 1.053 | 0.961 | 2.095 | 2.068 |
| fav...stores_1D | 0.916 | 0.989 | 0.968 | 0.949 | 1.036 | 0.970 | 0.980 | 1.032 | 1.197 | 1.238 |
| fav...stores_1M | 1.794 | 1.923 | 1.856 | 1.998 | 2.009 | 1.934 | 2.091 | 2.087 | 1.943 | 1.942 |
| fav...stores_1W | 2.024 | 2.054 | 2.046 | 1.968 | 2.128 | 2.123 | 2.197 | 2.101 | 2.220 | 2.357 |
| fav...trans_1D | 0.685 | 1.283 | 1.031 | 0.975 | 0.975 | 1.225 | 0.975 | 0.975 | 1.185 | 1.181 |
| fav...trans_1M | 0.943 | 1.214 | 1.089 | 1.133 | 1.397 | 1.244 | 1.390 | 1.358 | 1.152 | 1.179 |
| fav...trans_1W | 1.228 | 1.579 | 1.384 | 1.428 | 1.557 | 1.912 | 1.463 | 1.428 | 1.559 | 1.647 |
| fred_md_2025/cee | 3.468 | 4.823 | 3.349 | 4.490 | 4.490 | 3.873 | 4.490 | 4.490 | 3.745 | 3.643 |
| fred_md/macro | 5.680 | 6.623 | 5.307 | 5.842 | 5.842 | 6.399 | 5.842 | 5.842 | 5.743 | 5.794 |
| fred_qd_2025/cee | 2.192 | 2.455 | 2.046 | 2.181 | 1.773 | 2.292 | 2.296 | 2.365 | 1.903 | 2.123 |
| fred_qd/macro | 3.537 | 4.040 | 3.530 | 3.593 | 3.402 | 4.240 | 3.616 | 3.654 | 3.615 | 3.904 |
| gvar | 0.578 | 0.594 | 0.577 | 0.590 | 0.576 | 0.674 | 0.593 | 0.596 | 0.590 | 0.593 |
| hermes | 0.609 | 0.619 | 0.651 | 0.618 | 0.985 | 0.705 | 0.704 | 0.675 | 1.416 | 1.673 |
| hier...sales_1D | 0.557 | 0.552 | 0.547 | 0.552 | 0.547 | 0.572 | 0.551 | 0.551 | 0.720 | 0.793 |
| hier...sales_1W | 0.616 | 0.625 | 0.621 | 0.618 | 0.637 | 0.637 | 0.637 | 0.637 | 0.746 | 10.477 |
| hospital | 0.686 | 0.673 | 0.688 | 0.680 | 0.733 | 0.696 | 0.697 | 0.697 | 0.697 | 0.726 |
| hosp...sions_1D | 0.554 | 0.554 | 0.555 | 0.556 | 0.555 | 0.562 | 0.556 | 0.556 | 0.557 | 0.556 |
| hosp...sions_1W | 0.576 | 0.581 | 0.585 | 0.580 | 0.598 | 0.581 | 0.586 | 0.587 | 0.579 | 0.578 |
| jena_weather_10T | 0.354 | 0.398 | 0.389 | 0.357 | 0.368 | 0.413 | 0.418 | 0.418 | 0.673 | 0.742 |
| jena_weather_1D | 1.111 | 1.143 | 1.072 | 1.090 | 1.112 | 1.155 | 1.075 | 1.075 | 1.339 | 1.664 |
| jena_weather_1H | 0.353 | 0.429 | 0.356 | 0.359 | 0.362 | 0.413 | 0.367 | 0.367 | 0.452 | 0.553 |
| kdd_cup_2022_10T | 0.425 | 0.456 | 0.533 | 0.533 | 0.533 | 0.555 | 0.533 | 0.533 | 0.777 | 0.747 |
| kdd_cup_2022_1D | 0.704 | 0.709 | 0.697 | 0.698 | 0.704 | 0.715 | 0.708 | 0.709 | 0.730 | 0.751 |
| kdd_cup_2022_30T | 0.439 | 0.459 | 0.432 | 0.505 | 0.429 | 0.543 | 0.427 | 0.561 | 0.679 | 0.772 |
| m5_1D | 0.722 | 0.720 | 0.714 | 0.729 | 0.729 | 1.254 | 0.729 | 0.729 | 1.254 | 0.853 |
| m5_1M | 0.977 | 0.986 | 0.974 | 0.980 | 1.044 | 1.002 | 0.996 | 1.000 | 1.022 | 1.108 |
| m5_1W | 0.900 | 0.904 | 0.903 | 0.917 | 0.905 | 0.928 | 0.907 | 0.917 | 0.936 | 0.953 |
| proenfo_gfc12 | 0.649 | 0.614 | 0.908 | 0.917 | 0.917 | 0.834 | 0.917 | 0.917 | 1.305 | 2.431 |
| proenfo_gfc14 | 0.430 | 0.426 | 0.721 | 0.767 | 0.767 | 0.515 | 0.767 | 0.767 | 0.906 | 1.110 |
| proenfo_gfc17 | 0.485 | 0.528 | 0.889 | 0.900 | 0.900 | 0.672 | 0.900 | 0.900 | 1.142 | 2.135 |
| redset_15T | 0.790 | 1.208 | 0.833 | 0.741 | 0.818 | 1.250 | 1.041 | 1.243 | 1.231 | 1.231 |
| redset_1H | 1.365 | 1.338 | 1.337 | 1.367 | 1.306 | 1.321 | 1.410 | 2.279 | 1.859 | 2.377 |
| redset_5T | 0.654 | 0.749 | 0.787 | 0.723 | 0.719 | 0.711 | 0.793 | 1.026 | 2.690 | 1.224 |
| restaurant | 0.685 | 0.686 | 0.682 | 0.677 | 0.704 | 0.693 | 0.689 | 0.689 | 0.709 | 1.021 |
| rohlik_orders_1D | 0.959 | 1.052 | 0.986 | 1.006 | 1.135 | 1.341 | 0.970 | 1.051 | 1.211 | 1.447 |
| rohlik_orders_1W | 1.300 | 1.415 | 1.300 | 1.328 | 1.493 | 1.524 | 1.532 | 1.428 | 1.398 | 1.419 |
| rohlik_sales_1D | 0.881 | 0.899 | 1.148 | 1.096 | 1.218 | 1.375 | 1.170 | 1.147 | 1.248 | 1.266 |
| rohlik_sales_1W | 1.274 | 1.159 | 1.425 | 1.401 | 1.505 | 1.221 | 1.516 | 1.522 | 1.646 | 14.453 |
| rossmann_1D | 0.283 | 0.245 | 0.539 | 0.502 | 0.568 | 0.232 | 0.527 | 0.525 | 0.578 | 0.594 |
| rossmann_1W | 0.308 | 0.256 | 0.482 | 0.495 | 0.494 | 0.254 | 0.497 | 0.487 | 0.501 | 0.518 |
| solar_1D | 0.594 | 0.601 | 0.614 | 0.618 | 0.622 | 0.615 | 0.637 | 0.635 | 0.653 | 0.656 |
| solar_1W | 0.895 | 0.924 | 1.121 | 1.096 | 1.392 | 0.870 | 1.658 | 0.940 | 1.296 | 1.212 |
| s...weather_15T | 0.677 | 0.671 | 0.846 | 0.906 | 0.784 | 0.747 | 0.839 | 0.809 | 1.194 | 2.529 |
| s...weather_1H | 0.767 | 0.660 | 0.900 | 0.815 | 0.876 | 0.701 | 0.907 | 0.816 | 1.458 | 2.182 |
| uci...ality_1D | 1.046 | 1.147 | 1.128 | 1.205 | 1.260 | 1.186 | 1.138 | 1.092 | 1.123 | 1.181 |
| uci...ality_1H | 0.798 | 0.934 | 0.865 | 0.877 | 0.870 | 0.931 | 0.945 | 0.899 | 1.561 | $>10^3$ |
| uk_nat_1D/cum | 7.826 | 10.394 | 7.653 | 7.051 | 6.188 | 13.045 | 6.763 | 8.157 | 7.712 | 7.184 |
| uk_nat_1D/new | 2.037 | 2.071 | 1.992 | 2.135 | 2.039 | 2.076 | 2.135 | 2.122 | 2.799 | 2.741 |
| uk_nat_1W/cum | 2.783 | 3.478 | 3.192 | 4.011 | 2.824 | 2.872 | 3.014 | 3.435 | 2.238 | 2.399 |
| uk_nat_1W/new | 4.968 | 4.784 | 4.532 | 3.783 | 5.098 | 4.143 | 3.873 | 4.148 | 5.741 | 5.024 |
| uk_utla_1D/new | 3.725 | 3.815 | 3.729 | 3.512 | 4.036 | 3.801 | 3.565 | 3.531 | 5.582 | 5.623 |
| uk_utla_1W/cum | 17.442 | 18.932 | 19.435 | 18.486 | 16.286 | 16.912 | 19.325 | 17.489 | 14.331 | 16.313 |
| us_cons_1M | 1.464 | 1.698 | 1.467 | 1.605 | 1.564 | 1.571 | 1.513 | 1.516 | 1.486 | 1.445 |
| us_cons_1Q | 1.724 | 2.302 | 1.803 | 1.927 | 1.707 | 2.673 | 1.796 | 1.764 | 1.908 | 1.886 |
| us_cons_1Y | 3.730 | 4.807 | 3.634 | 4.007 | 3.898 | 4.180 | 4.807 | 4.108 | 3.786 | 4.081 |
| walmart | 0.648 | 0.696 | 0.707 | 0.679 | 0.907 | 0.662 | 0.845 | 0.774 | 1.217 | $>10^3$ |
| world_co2_emis | 2.670 | 2.761 | 2.643 | 2.876 | 2.716 | 2.720 | 2.875 | 2.754 | 2.688 | 7.724 |
| world_life_exp | 1.187 | 1.190 | 1.109 | 1.210 | 1.639 | 1.149 | 1.785 | 1.345 | 1.305 | 1.302 |
| world_tourism | 3.052 | 3.149 | 3.052 | 3.562 | 3.208 | 2.795 | 3.264 | 3.164 | 2.552 | 2.882 |

# I TabPFN Use Case Overview

Previous TabPFN models have been applied to a broad set of use cases. We now list 202 published use cases across different industries.

## Highlights

We highlight a selection of representative use cases that demonstrate TabPFN's strengths across domains:

1. TabPFN enabled non-invasive early detection of pancreatic cancer by integrating NMR metabolomics with clinical and protein biomarkers [77]. Link
2. TabPFN provided highly accurate predictions of donor mobilization success using baseline and post-mobilization variables, facilitating early triage and improved transplantation outcomes [78]. Link
3. TabPFN was used for effective differentiation between psychotic and non-psychotic major depression, improving classification accuracy and supporting psychiatric diagnosis [79]. Link
4. TabPFN served as a high-fidelity surrogate model for optimizing geopolymer concrete mix design, achieving superior accuracy, generalization, and low-uncertainty predictions compared to other ML approaches [80]. Link
5. TabPFN enabled robust prediction of silica nanoparticle cytotoxicity [81]. Link
6. TabPFN demonstrated superior performance and translational feasibility for liver fibrosis staging [82]. Link
7. TabPFN enables accurate prediction of reaction kinetics, facilitating mechanistic understanding in biochar-catalyzed antibiotic degradation processes [83]. Link
8. TabPFN serves as the top-performing regression model to estimate degradation kinetics from multi-source experimental data [83]. Link
9. TabPFN was employed as a core modeling component for learning from multimodal tabular data under strict temporal constraints, enabling strong discriminative performance, improved probability calibration, and effective causal forecasting in early rug-pull detection [84]. Link
10. TabPFN was fine-tuned into a domain-specific model (FinPFN) for regime-aware stock return prediction, improving performance in non-stationary financial markets by adapting to evolving feature–return relationships [85]. Link
11. TabPFN enabled early fault classification in rotating machinery, addressing data scarcity in industrial scenarios [86]. Link

## Healthcare and Life Sciences

We collected 98 published TabPFN use cases in this area. Applications span diagnosis, prognosis, treatment response prediction, and biomarker-based modeling under frequent data scarcity.

1. TabPFN enabled non-invasive early detection of pancreatic cancer by integrating NMR metabolomics with clinical and protein biomarkers. [77]. Link
2. TabPFN enables highly accurate and cost-efficient molecular property prediction by pairing in-context learning with frozen molecular embeddings and descriptor [87]. Link
3. TabPFN enabled robust prediction of silica nanoparticle cytotoxicity [81]. Link
4. TabPFN was combined with BulkFormer to improve prediction accuracy of post-transplant kidney function for better assessment of organ viability during machine perfusion or cold storage [88]. Link

5. TabPFN enhances survival analysis, leading to superior performance compared to specialized methods [89]. Link

6. TabPFN demonstrated superior performance and translational feasibility for liver fibrosis staging [82]. Link

7. TabPFN was leveraged in cardiovascular disease diagnosis [90]. Link

8. TabPFN enabled accurate prediction of ALM from multimodal clinical data and improved sarcopenia screening by maintaining robust performance despite missing modalities [91]. Link

9. TabPFN was employed in the winning solution for predicting walking function [92]. Link

10. TabPFN demonstrated high accuracy and specificity in matching cell line transcriptomes to reference kidney cell types using curated kidney marker gene lists, enhancing robust assessment of cell line identity [93]. Link

11. TabPFN was used to enhance prediction accuracy of protein coupling based on structural features, improving biological insight into protein interactions [94]. Link

12. TabPFN supports risk stratification and adverse event prediction in chemotherapy-based stem cell mobilization, enabling improved ward management and resource allocation [95]. Link

13. TabPFN used with other ML models to improve radiomics-based breast cancer diagnosis, enhancing feature-combination performance and classification accuracy [96]. Link

14. TabPFN enhances model interpretability and accuracy in differentiating complex spinal infections, aiding clinical decision-making in ambiguous diagnostic cases [97]. Link

15. TabPFN enables improved data quality and predictive model reliability by integrating unstructured clinical text with automated pipelines, enhancing early disease prediction and clinical decision-making [98]. Link

16. TabPFN improved severity classification performance in diabetic retinopathy, supporting more accurate staging and treatment planning [99]. Link

17. TabPFN was integrated into the multimodal MuCB-tabpfn framework, enabling high predictive accuracy in estimating pollutant concentrations in human blood [100]. Link

18. TabPFN enables better generalization and accuracy in modeling complex drug formulation data, improving AI-driven formulation design workflows [101]. Link

19. TabPFN enables state-of-the-art real-time stress detection by enhancing accuracy and interpretability of multimodal physiological and sensor data [102]. Link

20. TabPFN was applied as a robust and data-efficient alternative for tabular learning in drug discovery, improving performance on small and medium datasets and under out-of-distribution conditions [103]. Link

21. TabPFN was used to enhance clinical risk prediction from electronic health records by providing robust modeling under real-world constraints, improving prognosis accuracy and reliability [104]. Link

22. TabPFN achieved the highest performance in predicting BCRL risk with strong minority-class discrimination and accurate calibration [105]. Link

23. TabPFN achieved strong generalization performance in predicting adsorption capacity in zeolites, with physically meaningful interpretability [106]. Link

24. TabPFN achieved superior discriminative performance in predicting RSA risk by integrating multidimensional clinical data into accurate and interpretable screening models [107]. Link

25. TabPFN was used to encode structured EHR data for predicting peak $VO_2$ and identifying high-risk heart failure patients [108]. Link

26. TabPFN provided highly accurate predictions of donor mobilization success using baseline and post-mobilization variables, facilitating early triage and improved transplantation outcomes [78]. Link

27. TabPFN was integrated into the FocalTab framework to improve classification accuracy, handle class imbalance, and support early identification of adolescent alcohol use [109]. Link

28. TabPFN demonstrated strong robustness in cross-cohort microbiome disease prediction under domain shift, maintaining competitive performance across datasets [110]. Link

29. TabPFN was used as a meta-learner combining predictions of multiple base models to capture complex interactions and enhance early coronary artery disease prediction accuracy [111]. Link

30. TabPFN enables Bayesian inference via in-context learning without per-dataset training, improving accuracy, calibration, and inference speed in scientific disease modeling tasks [112]. Link

31. TabPFN was extended to multimodal learning through MMPFN, enabling effective integration of non-tabular modalities with structured clinical data [113]. Link

32. TabPFN enables unified Bayesian modeling to improve bioactivity prediction across the ChEMBL database, supporting more efficient drug discovery pipelines [114]. Link

33. TabPFN was used for effective differentiation between psychotic and non-psychotic major depression, improving classification accuracy and supporting psychiatric diagnosis [79]. Link

34. TabPFN enables more accurate and efficient causal inference to aid early diagnosis and understanding of Long COVID [115]. Link

35. TabPFN was utilized to improve clinical risk prediction models on MIMIC-III data, enhancing both accuracy and efficiency [116]. Link

36. TabPFN outperformed current methods in predicting HFNC therapy outcomes and demonstrated potential for improved performance with additional clinical measurements [117]. Link

37. TabPFN was used in a hybrid model combining radiomics and deep learning features to improve risk stratification for post-TIPS hepatic encephalopathy [118]. Link

38. TabPFN was fine-tuned as a proxy model to predict synthetic likelihood of hMOFs, enabling high-fidelity large-scale screening in materials-related biomedical contexts [119]. Link

39. TabPFN improved intra-European ancestry prediction accuracy when combined with ML-based marker selection, outperforming traditional approaches [120]. Link

40. TabPFN improves renal tumor classification accuracy in CT radiomics by effectively handling small, high-dimensional datasets without extensive tuning [121]. Link

41. TabPFN demonstrates competitive performance as a count-based model for clinical prediction on structured EHR data compared to transformer-based pipelines [122]. Link

42. TabPFN improves empathy detection accuracy and cross-subject generalization in human-centered video datasets [123]. Link

43. TabPFN enables accurate prediction of reaction kinetics, facilitating mechanistic understanding in biochar-catalyzed antibiotic degradation processes [83]. Link

44. TabPFN yields competitive or superior performance for multiple imputation tasks compared to alternative statistical and ML methods [124]. Link

45. TabPFN improves multimodal skin cancer diagnosis by combining structured lesion features with clinical data for more accurate and interpretable predictions [125]. Link

46. TabPFN supports pediatric disease classification in clinical decision support systems, reducing misdiagnosis in emergency settings [125]. Link

47. TabPFN improves EEG seizure classification across subjects, achieving high accuracy and strong generalization [126]. Link

48. TabPFN improves kelp origin prediction using stable isotope data, providing robust and interpretable environmental insights [127]. Link

49. TabPFN predicts $CO_2$ frosting temperatures in natural gas mixtures with high accuracy and interpretability [128]. Link

50. TabPFN improves ADMET modeling by increasing prediction accuracy, simplifying deployment, and producing compact models [129]. Link

51. TabPFN enhances analysis and classification of volatile organic compounds using mass spectrometry data, improving efficiency in chemical and biomedical analysis [130]. Link

52. TabPFN was applied to distinguish cancer patients from healthy individuals using immune system profiles from peripheral blood, facilitating predictions of immunotherapy responses [131]. Link

53. A machine learning model employing TabPFN was developed for non-invasive diagnostic prediction of minimal change disease in patients with nephrotic syndrome, utilizing clinical biomarkers [132]. Link

54. TabPFN was integrated into a system for analyzing T-cell receptor repertoires combined with clinical biomarkers to forecast immunotherapy outcomes in cancer patients, as explored by researchers at BostonGene [133]. Link

55. TabPFN enabled early detection of stillbirth risks through analysis of cardiotocography data, supporting improved prenatal care [134]. Link

56. Predictive modeling for postoperative outcomes following anterior cervical corpectomy utilized TabPFN to assess patient demographics and surgical parameters [135]. Link

57. A hybrid model incorporating TabPFN was introduced to predict dementia progression in Parkinson's disease patients, handling small datasets and missing values effectively [136]. Link

58. A machine learning model based on TabPFN was developed to predict 90-day unfavorable outcomes in stroke patients with distal vessel occlusions using CT perfusion imaging [137]. Link

59. TabPFN facilitated the prediction of non-invasive ventilation outcomes in patients with acute hypoxemic respiratory failure, supporting early identification of treatment failures [138]. Link

60. An interpretable Transformer-based model leveraging TabPFN was created to predict intravenous immunoglobulin resistance in pediatric patients with Kawasaki disease [139]. Link

61. TabPFN was employed in visual representation techniques for prostate cancer diagnosis, converting clinical biomarkers and symptom data into formats suitable for analysis [140]. Link

62. TabPFN was used to combine clinical, MR morphological, and delta-radiomics features to predict lymphovascular invasion in invasive breast cancer patients [141]. Link

63. TabPFN is proposed to predict mental health trajectories through digital phenotyping, enabling proactive and personalized interventions in precision psychiatry [142]. Link

64. TabPFN contributed to cardiovascular disease risk stratification using clinical features from a large patient cohort, incorporating interpretability techniques [143]. Link

65. TabPFN outperformed traditional machine learning models for early prediction of acute kidney injury in hospitalized patients, demonstrating generalizability across datasets [144]. Link

66. TabPFN was integrated into a framework for predicting postoperative mobility and discharge destinations in older adults using sensor data [145]. Link

67. TabPFN supported the prediction of infant temperament from maternal mental health data, aiding early identification of at-risk infants [146]. Link

68. TabPFN was employed to characterize clinical risk profiles for complications in type 2 diabetes mellitus patients, focusing on neuropathy and retinopathy [147]. Link

69. TabPFN was extended with a longitudinal-to-cross-sectional transformation to forecast Alzheimer's disease progression on neuroimaging datasets [148]. Link

70. TabPFN supported uncertainty calibration evaluation in medical data using variational techniques [149]. Link

71. TabPFN was applied to predict tumor response to chemotherapy in cholangiocarcinoma patients using RNA expression landscapes [150]. Link

72. TabPFN was incorporated into a generative model framework for tasks like data augmentation and imputation in biomedicine [151]. Link

73. TabPFN facilitated the prediction of gallstone malignancy risks through analysis of associated disease factors [152]. Link

74. TabPFN was used in classifying tuberculosis treatment outcomes based on clinical and sociodemographic data from national registries [153]. Link

75. TabPFN contributed to early prediction of gestational diabetes using cell-free DNA and genetic scores from early pregnancy blood samples [154]. Link

76. TabPFN was used for predicting schizophrenia based on sense of agency features, emphasizing interpretability [155]. Link

77. TabPFN was integrated into a physiologically based pharmacokinetic model for predicting dissolution and absorption of amorphous solid dispersions in drug development [156]. Link

78. TabPFN enabled classification of respiratory diseases from sound data, addressing clinical spectrum diversity [157]. Link

79. TabPFN was applied to small-data tabular learning in drug discovery, handling data scarcity and distribution shifts [158]. Link

80. TabPFN facilitated prediction of coronary heart disease risk in patients with cardiovascular-kidney-metabolic syndrome, optimizing evaluation in small samples [159]. Link

81. TabPFN was used to predict success of allogeneic stem cell mobilization in donors, aiding transplant therapies [160]. Link

82. TabPFN contributed to predicting manual strength using anthropometric data, focusing on accuracy and interpretability [161]. Link

83. TabPFN supported uncertainty-guided model selection for biomolecule efficacy prediction, enhancing ensemble optimization in drug discovery, as studied at GSK [162]. Link

84. TabPFN was utilized in a multitask deep learning framework for optimizing in vitro fertilization decisions, including embryo transfer and pregnancy prediction [163]. Link

85. TabPFN enabled a framework for early Long COVID detection through causal gene identification and interpretability [164]. Link

86. TabPFN was used for neoadjuvant therapy recommendations in breast cancer, integrating multi-omics data [165]. Link

87. TabPFN facilitated prediction of recurrence and progression in oral potentially malignant disorder patients post-surgery [166]. Link

88. TabPFN supported prediction of occult lymph node metastasis in non-small cell lung cancer patients treated with stereotactic ablative radiotherapy [167]. Link

89. TabPFN was used in stroke diagnosis, addressing dataset imbalance and model interpretability for clinical decisions [168]. Link

90. TabPFN was used to predict diabetes-related hypo- and hyperglycemia during hemodialysis using continuous glucose monitoring data, facilitating improved patient management [169]. Link

91. TabPFN was applied to enhance diagnosis of hypervascular thyroid nodules using multimodal ultrasound features [170]. Link

92. TabPFN was integrated with radiomics and clinical features to predict endovascular treatment success in femoropopliteal chronic total occlusions, supporting interventional planning [171]. Link

93. TabPFN was applied to CorvisST biomechanical indices to classify corneal disorders, improving diagnostic accuracy in ophthalmology [172]. Link

94. TabPFN was incorporated into a non-invasive sleep staging framework using respiratory sound features, advancing passive sleep monitoring [173]. Link

95. TabPFN supported prediction of vancomycin blood concentrations to optimize antimicrobial dosing strategies in clinical practice [174]. Link

96. TabPFN was used to predict negative self-rated oral health in adults, identifying risk factors for targeted public-health interventions [175]. Link

97. TabPFN was extended to very high-dimensional feature spaces to enable robust analysis of biomedical data, improving stability and interpretability in clinical applications [176]. Link

98. TabPFN predicted gastrointestinal bleeding risk in pediatric Henoch–Schönlein purpura patients, supporting early clinical intervention [177]. Link

## Financial Services, Banking, and Insurance

We collected 7 published TabPFN use cases in this area. These applications include risk modeling, actuarial analysis, credit-related prediction, and customer analytics.

1. TabPFN improves low-supervision transaction analytics by doubling zero-shot MCC on churn prediction and enhancing few-shot MCC, enabling better knowledge-grounded reasoning in financial transaction analysis [178]. Link

2. TabPFN serves as a strong tabular baseline for financial transaction analytics (e.g., churn prediction) [179]. Link

3. TabPFN was employed as a core modeling component for learning from multimodal tabular data under strict temporal constraints, enabling strong discriminative performance, improved probability calibration, and effective causal forecasting in early rug-pull detection [84]. Link

4. TabPFN was used to predict forward financial returns, aiding investment strategy evaluation with the adjusted Sharpe ratio to enhance financial forecasting accuracy [180]. Link

5. TabPFN was fine-tuned into a domain-specific model (FinPFN) for regime-aware stock return prediction, improving performance in non-stationary financial markets by adapting to evolving feature–return relationships [85]. Link

6. TabPFN was benchmarked against leading AutoML frameworks on financial classification tasks, demonstrating strong performance in multiclass settings [181]. Link

7. TabPFN facilitated cross-selling of health insurance products through deep learning analysis of customer data [182]. Link

## Energy and Utilities

We collected 24 published TabPFN use cases in this area. They include environmental forecasting, renewable-energy prediction, and process or asset optimization across energy and utility systems.

1. TabPFN was used as a surrogate model for fast one-step predictions under irregular measurements, aiding the delay-aware digital twin framework in handling nonlinear dynamics and operational delays in biogas production control [183]. Link

2. TabPFN provided superior fitting performance for models analyzing biochar's impact on soil cadmium contamination, improving prediction accuracy in artificial and natural aging scenarios [184]. Link

3. TabPFN was used to improve the robustness and accuracy of photovoltaic power forecasting models by providing unified in-context prediction and strong generalization with heterogeneous inputs [185]. Link

4. TabPFN enables effective learning and prediction with very limited data by leveraging pretrained tabular inference, improving model performance in challenging geological prediction tasks [186]. Link

5. TabPFN was used as a baseline for comparison in spatiotemporal forecasting of small Earth data, demonstrating value despite being surpassed in accuracy and robustness by the proposed method [187]. Link

6. TabPFN demonstrated superior predictive performance under sparse sampling conditions, enabling accurate high-resolution mapping of groundwater bicarbonate concentrations and evaluation of scaling risks [188]. Link

7. TabPFN was used for slope stability assessment, providing superior accuracy and robustness with limited sample sizes and enhancing regional scale evaluation efficiency [189]. Link

8. TabPFN surpasses other models in solar energy meteorology [190]. Link

9. TabPFN Regression was used as a predictive model for evaluating trophic level index from multi-source remote sensing data within the modeling framework [191]. Link

10. TabPFN-based data augmentation improved model robustness under limited data, enabling accurate predictions of electrochemical performance and efficient screening of hard carbon candidates [192]. Link

11. TabPFN was employed to predict river algal blooms through multi-classification of chlorophyll-a concentrations, aiding water management [193]. Link

12. TabPFN facilitated wildfire propagation prediction in Canadian conifer forests, classifying fire types for environmental risk assessment [194]. Link

13. TabPFN was integrated into a machine learning framework for optimizing energy consumption at wastewater treatment plants [195]. Link

14. TabPFN supported rainfall forecast post-processing using historical error patterns from environmental data [196]. Link

15. TabPFN enabled solar forecast error adjustment, particularly during rapid weather changes, as developed by Open Climate Fix [197]. Link

16. TabPFN was applied to predict ash fusibility in high-alkali coal for improved energy production [198]. Link

17. TabPFN contributed to predicting Henry coefficients for alkanes in zeolites, aiding hydroisomerization in sustainable fuel production [199]. Link

18. TabPFN facilitated shape-selectivity modeling in zeolites for long-chain alkane hydroisomerization, optimizing catalyst design [200]. Link

19. TabPFN was used in an integrated framework for estimated ultimate recovery prediction and fracturing optimization in shale gas reservoirs [201]. Link

20. TabPFN supported core data augmentation for enhanced reservoir parameter prediction in oil and gas exploration [202]. Link

21. TabPFN was employed to optimize energy performance in multistage centrifugal pumps through entropy generation analysis [203]. Link

22. TabPFN was applied to generate advanced global heat flow maps at 0.2° resolution, integrating high-resolution geophysical data to improve geothermal resource modeling [204]. Link

23. TabPFN contributed to FuelCast, standardizing benchmarks for ship fuel consumption prediction and improving efficiency in maritime operations [205]. Link

24. TabPFN was used as the main supervised classifier to automatically identify thunderstorm ground enhancements from particle detector and environmental measurements [206]. Link

## Industrial and Manufacturing

We collected 41 published TabPFN use cases in this area. These applications cover industrial prediction, process optimization, and engineering-related modeling tasks.

1. TabPFN served as a high-fidelity surrogate model for optimizing geopolymer concrete mix design, achieving superior accuracy, generalization, and low-uncertainty predictions compared to other ML approaches [80]. Link

2. TabPFN enables rapid prediction of structural crack behavior, supporting reliability assessment and failure analysis in ultra-high-performance concrete [207]. Link

3. TabPFN leveraged prior-data pretraining to predict WCFZ height from only 76 field samples without extensive tuning, providing superior and generalizable performance compared to other ML models [208]. Link

4. TabPFN's multitask-aware prior adaptation improves predictive accuracy and computational efficiency in steel property prediction, enabling scalable, rapid, and reliable deployment for industrial quality control and process optimization [209]. Link

5. TabPFN's pre-trained foundation model enables strong small-data regression and well-calibrated uncertainty estimates in a single forward pass, significantly reducing evaluation cycles for active learning in materials discovery [210]. Link

6. TabPFN demonstrated strong generalization ability in predicting crash severity, contributing to improved data-driven safety interventions in electric vehicle crash contexts [211]. Link

7. TabPFN excelled in zero-shot inference and robustness for rare crash categories, enhancing classification of uncommon SAE automation levels with limited data [212]. Link

8. TabPFN 2.5's dataset-level embedding identified 'engineering-like' synthetic datasets to enable continued pre-training on synthetic tasks, significantly improving accuracy and data efficiency over baseline models and AutoGluon on engineering regression datasets [213]. Link

9. TabPFN achieved the highest prediction accuracy in predicting concrete fracture properties and, combined with SHAP analysis, provided detailed and unbiased insights into nonlinear and interaction effects [214]. Link

10. TabPFN significantly reduces computational overhead and data requirements while enabling rapid, flexible, and data-efficient engineering design with competitive diversity and low performance error in generated designs [215]. Link

11. TabPFN served as a backbone combined with graph neural network embeddings and MagpieEX descriptors for effective, data-efficient, and physics-aware materials property prediction, outperforming sophisticated models [216]. Link

12. TabPFN was used for spatial predictions and imputations in geotechnical modeling, achieving superior accuracy, faster inference, and well-calibrated predictive distributions compared to hierarchical Bayesian baselines [217]. Link

13. TabPFN provided strong prediction ability, outperforming alternatives and enabling more accurate performance prediction of biochar-modified concrete [218]. Link

14. TabPFN was used for accurate and reliable monitoring of driver alertness levels in challenging driving environments, proving more effective than traditional models like logistic regression and XGBoost [219]. Link

15. TabPFN enabled highly accurate and unbiased prediction of RAC's elastic modulus, improving trustworthiness and interpretability in a challenging heterogeneous materials domain [220]. Link

16. TabPFN provided meta-learned prior knowledge that enhanced predictive performance and uncertainty quantification in the PSF-Net model for reliable 5G RF-EMF exposure assessment [221]. Link

17. TabPFN showed superior predictive performance in predicting the hardgrove grindability index, improving model accuracy [222]. Link

18. TabPFN delivered the best overall performance with the lowest error metrics and highest $R^2$ and composite score, demonstrating superior predictive capability for asphalt concrete strength [223]. Link

19. TabPFN was applied to efficient multi-objective optimization of non-linear mixture designs, improving strength, reducing costs, and lowering carbon emissions for sustainable mining applications [224]. Link

20. TabPFN was employed for highly accurate and statistically superior predictions of pavement roughness by capturing complex interactions among traffic loads, structural parameters, and climatic factors [225]. Link

21. TabPFN enables accurate prediction of CPB strength with limited data, improving efficiency and supporting theoretical understanding and practical application in mining industry tailings management [226]. Link

22. TabPFN's improved spatiotemporal architecture enhances robustness and accuracy in geological condition detection, enabling better multi-step predictions with uncertainty quantification in tunnel construction [227]. Link

23. TabPFN was utilized as a core component in a multi-objective optimization framework to design cemented foam backfill optimizing high strength, low cost, and low carbon emissions [224]. Link

24. TabPFN enhances prediction accuracy and reliability with small sample sizes and missing features in geotechnical engineering [186]. Link

25. TabPFN enabled interpretable and uncertainty-aware parameter inference, improving predictions and revealing geotechnical relationships without model retraining for data-scarce applications [228]. Link

26. TabPFN was used to accurately predict compressive strength in geopolymer concrete from small datasets, supporting optimization of material composition and process parameters in construction material science [229]. Link

27. TabPFN was used to improve prediction accuracy in concrete property estimation by integrating knowledge-constrained data augmentation [230]. Link

28. TabPFN enabled efficient and accurate mapping of key leaf-vein texture parameters to lubrication performance metrics, facilitating multi-objective optimization to identify optimal texture designs that improve journal bearing performance [231]. Link

29. TabPFN enables robust mapping between operating boundary conditions and latent features to manage data scarcity and enhance regression accuracy, resulting in faster and more accurate temperature field reconstruction [232]. Link

30. TabPFN enables encoding of structured device-physics primitives for reliable and precise analog circuit optimization, outperforming Gaussian-process methods in sample efficiency and final metric quality [233]. Link

31. TabPFN enabled early fault classification in rotating machinery, addressing data scarcity in industrial scenarios [86]. Link

32. TabPFN facilitated microcontroller performance prediction, aiding semiconductor screening with minimal supervision, as studied at Infineon Technologies [234]. Link

33. TabPFN was applied to caisson inclination prediction in ultra-deep construction, combining data denoising techniques [235]. Link

34. TabPFN supported event classification in phase-sensitive optical time-domain reflectometry systems for distributed fiber sensing [236]. Link

35. TabPFN was integrated into an adaptive ensemble for intrusion detection in Industrial Internet of Things networks [237]. Link

36. TabPFN enabled a random forest-based framework for attack recognition in Internet of Things networks, improving interpretability [238]. Link

37. TabPFN was used in cryogenic-assisted abrasive waterjet machining for improving surface integrity in titanium alloys [239]. Link

38. TabPFN supported in-context learning for thermal behavior prediction in nano-phase change materials for battery systems [240]. Link

39. TabPFN was applied to explainable strength evaluation in multicomponent concrete mixtures [241]. Link

40. TabPFN was integrated into a multimodal fusion framework linking microstructure to friction behavior in martensitic stainless steel, improving wear resistance in materials engineering applications [242]. Link

41. TabPFN supported multiscale modeling to predict soil salinity in arid farmland, advancing sustainable agricultural management in regions such as Xinjiang [243]. Link

## Other Industries

We collected 32 further published TabPFN use cases in this area, spanning a heterogeneous set of domains and prediction tasks.

1. TabPFN enables the construction of credal sets for models where it was previously infeasible, broadening uncertainty representation and improving uncertainty estimation [244]. Link

2. TabPFN enables efficient and valid hypothesis testing for feature relevance in tabular data, allowing accurate statistical inference in nonlinear and correlated settings [245]. Link

3. TabPFN enables efficient computation of conditional Shapley values, resulting in faster and often more accurate explainable AI analysis [246]. Link

4. TabPFN enables effective node classification by leveraging engineered tabular features from graph data as a practical and competitive alternative to graph-specific and language-based foundation models [247]. Link

5. TabPFN was integrated as the surrogate model enabling accurate and efficient prediction with uncertainty estimation, enhancing the performance, scalability, and zero-shot transfer capability of the DB-SAEA framework [248]. Link

6. TabPFN was used to model the relationship between nuclear structure properties and $\alpha$-particle preformation factors, improving $\alpha$-decay half-life predictions and enabling insights into nuclear shell effects and magic numbers [249]. Link

7. TabPFN served as the foundation for TabMGP, enabling state-of-the-art predictive capabilities with effective epistemic uncertainty quantification and improved posterior inference in tabular data contexts [250]. Link

8. TabPFN demonstrated superior utility for real-world operational yield forecasting due to faster tuning and reduced feature engineering requirements [251]. Link

9. TabPFN serves as the base learner in a multi-stage ensemble to model recognition probabilities of rural villages, enabling identification of high-potential but under-observed candidates in geospatial, highly imbalanced datasets [252]. Link

10. TabPFN was used as a base learner in a stacking ensemble model, improving prediction accuracy and performance for soil salinity retrieval from multispectral imagery data [253]. Link

11. TabPFN serves as the foundational model for ExplainerPFN, enabling zero-shot estimation of Shapley values for feature importance without access to the predictive model or reference explanations [254]. Link

12. TabPFN enables accurate classification of Near-Earth Objects as Potentially Hazardous, facilitating early identification and monitoring of potential asteroid threats [255]. Link

13. TabPFN improves malware detection performance in limited data scenarios by outperforming traditional ensemble models, enhancing cybersecurity workflows [256]. Link

14. TabPFN achieved the best performance in predicting mycotoxin contamination, outperforming baseline and transfer learning models to enhance prediction accuracy for early interventions [257]. Link

15. TabPFN was used in a classification pipeline whose latent space provided a 2D representation of the blazar population, revealing a continuum between blazar types [258]. Link

16. TabPFN enhances accuracy and efficiency in predicting grapevine diseases by processing complex environmental data and providing per-pixel disease probabilities for precise vineyard disease management [259]. Link

17. TabPFN enhances synthetic tabular data generation by providing probabilistic modeling capabilities that improve data quality, realism, and utility [260]. Link

18. TabPFN was modified for microbiome data classification in metagenomics, matching species abundance patterns with synthetic priors [261]. Link

19. TabPFN enabled lunar regolith analysis for classifying meteorite compositions from spectral data [262]. Link

20. TabPFN facilitated winter wheat yield forecasting in agricultural regions by integrating climate and remote sensing data [263]. Link

21. TabPFN was applied to flood impact assessment on housing prices by geographic areas [264]. Link

22. TabPFN showed the strongest performance on 31 predictive soil modeling datasets containing 30 to 460 samples [265]. Link

23. TabPFN was applied to shallow natural gas hazard prediction in tunnel construction [266]. Link

24. TabPFN supported automated feature engineering for energy consumption forecasting in domain-specific applications [267]. Link

25. TabPFN enabled Australian rice phenology prediction using remote sensing and weather data for crop management [268]. Link

26. TabPFN was applied to a multi-stage framework for predicting fuel blend properties through automated feature engineering [269]. Link

27. TabPFN enabled kriging prior regression for incorporating spatial context in soil mapping predictions [270]. Link

28. TabPFN enhanced clone-type recognition across programming languages through metrics-driven analysis, improving stability and interpretability in software engineering [271]. Link

29. TabPFN informed the development of TabImpute, enabling efficient zero-shot imputation for missing tabular data and improving preprocessing pipelines [272]. Link

30. TabPFN, alongside TabICL and related foundation models, was evaluated for intrusion detection, improving cybersecurity performance in IoT networks [273]. Link

31. TabPFN was used in forensic science to advance biogeographical ancestry predictions [274]. Link

32. TabPFN was used as a benchmark model for predicting avocado alternate bearing from Sentinel-2 and climate features [275]. Link