Beyond Accuracy on RelBench: Item Response Theory Analysis of Relational Deep Learning Benchmarks

Dinesh Katupputhur Ramprasath [1] **Tom Palczewski** [1] **Joe Meyer** [1] **Roshan Reddy Upendra** [1] **Minghua Li** [1]

Abstract

Machine learning benchmarks typically rank models by aggregate performance metrics, implicitly assuming all test instances are equally informative. We apply Item Response Theory (IRT) to two benchmark suites for relational deep learning, RelBench and DBInfer, to reveal properties of *benchmark structure*: which items drive rankings, where models diverge, and whether certain models are systematically over or under-ranked. Across 17 datasets (10 RelBench + 7 DBInfer, spanning 55 tasks), we evaluate up to 39 models from 13 families, including gradient-boosted decision trees, graph neural networks (HeteroGraphSAGE, GatedGNN), graph foundation models (RelGT, Griffin, GNN+TabPFN-2.5), single-table foundation models applied via flat-table features (TabPFN, TabICL, ContextTab, TabSTAR), relational DFS-based learners (RDBLearn), and AutoML (AutoGluon). We construct binary response matrices and fit Rasch, 2PL, and LogisticFM models with bootstrap resampling (N=100). Our key findings are: (1) IRT ability rankings correlate strongly with accuracy on 11 of 17 datasets ($\rho > 0.8$), but diverge substantially when models have nearly identical accuracy (Spearman ρ as low as 0.011 on dbinfer-diginetica); (2) graph-based models show strong domain-dependence—HeteroGraphSAGE ranks #1 by IRT on 4 datasets but drops to the bottom tercile on others; (3) RDBLearn's Deep Feature Synthesis improves IRT ability on some datasets (e.g., $\Delta\hat{\theta} = +0.286$ on rel-ratebeer) but hurts on others; (4) IRT flags datasets where near-zero accuracy spread makes aggregate rankings unreliable for model selection (e.g., dbinfer-diginetica and dbinfer-retailrocket); (5) binarization ablation studies confirm robustness of IRT findings. In summary, IRT serves as a diagnostic tool: it can confirm when rankings are stable, and flag cases where rankings are compressed, item-driven, or uninformative for model selection.

1. Introduction

The evaluation of machine learning models has predominantly relied on aggregate performance metrics (accuracy, AUC-ROC, F1 score) computed over held-out test sets. While simple and widely understood, these metrics implicitly assume that all test instances contribute equally to the evaluation signal. A model that excels on trivially easy instances but fails on genuinely challenging ones receives the same credit as one that masters the hard cases, potentially leading to misleading benchmark rankings.

This gap is especially consequential in relational deep learning, where benchmarks like RelBench (Robinson et al., 2024) standardize evaluation of models operating on multi-table relational databases. Recent work has combined single-table foundation models (TabPFN (Hollmann et al., 2023), TabICL (Ye et al., 2024), ContextTab (Marco Spinaci, 2025), TabSTAR (Alan Arazi, 2025)) with relational feature engineering systems such as RDBLearn (Zhang et al., 2025) and Deep Feature Synthesis (Kanter & Veeramachaneni, 2015) to achieve competitive performance on these benchmarks. Note that these foundation models are single-table learners; in our evaluation pipeline, they receive flat-table features extracted from the relational database via simple joins or, when wrapped by RDBLearn, via automatic Deep Feature Synthesis. Meanwhile, graph-based approaches, for example Griffin (Hu et al., 2025) and RelGT (Rishab et al., 2025) convert relational schemas into heterogeneous graphs. However, the structure of benchmark rankings, including which items drive them, where models diverge, and whether certain models are systematically over- or under-ranked, remains unexamined.

To the best of our knowledge, no prior work has quanti-

[1]SAP, Palo Alto, CA, USA. Correspondence to: Dinesh Katupputhur Ramprasath <dinesh.katupputhur.ramprasath@sap.com>, Tom Palczewski <tom.palczewski@sap.com>.

The Workshop on Graph Foundation Models at the 43rd International Conference on Machine Learning (ICML 2026), Seoul, South Korea.

fied the reliability of model rankings on relational benchmarks using a formal measurement-theoretic framework. Item Response Theory (IRT), originating in educational testing (Lord & Novick, 1968; Rasch, 1960), treats each test instance as an "item" and each model as an "examinee," jointly estimating latent ability θ_j for models and difficulty β_i for items. Unlike aggregate accuracy, IRT reveals *which items differentiate models* and *which models succeed on hard items*.

A preliminary analysis reveals a possible anomaly in benchmark quality: Dummy (majority-class) ranks #1 by accuracy on rel-stack (0.957) and rel-salt (0.693), ahead of most ML models, due to severe class imbalance. On dbinfer-diginetica, 37 models all achieve $\sim$0.990 accuracy, making the benchmark uninformative (IRT $\rho = 0.011$). Such artifacts are invisible to aggregate metrics but readily identified by IRT's discrimination parameters.

Our contributions are:

1. We conduct, to the best of our knowledge, the first IRT-based structural analysis of relational deep learning benchmarks, evaluating up to 39 models from 13 families across 17 datasets (55 tasks), with bootstrap resampling for statistical robustness.

2. We demonstrate that IRT reveals meaningful benchmark structure: ranking divergence (ρ ranging from 0.011 to 0.992) varies inversely with accuracy spread, and models exploiting easy items are systematically over-ranked by accuracy.

3. We reveal strong domain-dependence of graph-based models: HeteroGraphSAGE ranks #1 on rel-amazon, rel-arxiv, rel-trial, and dbinfer-stackexchange, but ranks last (35/35) on rel-avito and is absent on several other datasets.

4. We provide the first IRT comparison of RDBLearn (DFS-wrapped foundation models) against their standalone counterparts, showing that DFS helps on some datasets (up to $\Delta\hat{\theta} = +0.286$) but hurts on others ($-1.211$).

5. We conduct binarization ablation studies confirming robustness of IRT findings.

2. Related Work

IRT in Machine Learning. The application of psychometric models to ML evaluation traces back to Martínez-Plumed et al. (2019), who fitted IRT models to classification benchmarks. Vania et al. (2021) applied IRT to NLU benchmarks, finding many items with near-zero discrimination. Polo et al. (2024) showed that IRT-selected subsets of MMLU and HELM can approximate full rankings with $100\times$ fewer items.

Our work differs in several ways: (1) we target relational benchmarks rather than single-table tabular tasks or NLP tasks; (2) we include graph-based narrow and foundation models (HeteroGraphSAGE, GNN+TabPFN-2.5, GatedGNN, RelGT (Rishab et al., 2025), Griffin (Hu et al., 2025)) and relational DFS-based models that exploit multi-table structure; (3) we employ LogisticFM (Cagnone & Monari, 2013; Stanford CS321M, 2024) for multi-factor latent structure; and (4) we analyze ranking consistency across 17 datasets in a multi-task setting.

Relational Deep Learning. RelBench (Robinson et al., 2024) provides standardized relational database benchmarks with datasets spanning racing (rel-f1), e-commerce (rel-amazon), Q&A (rel-stack), and others. Models span flat-table approaches (XGBoost, LightGBM), graph-based approaches (HeteroGraphSAGE (Fey et al., 2023), RelGT (Rishab et al., 2025), Griffin (Hu et al., 2025)), and DFS-based learners (RDBLearn (Zhang et al., 2025)). We additionally evaluate on 7 DBInfer datasets.

Tabular Foundation Models. TabPFN (Hollmann et al., 2023) performs in-context learning for tabular classification. TabICL (Ye et al., 2024) extends this with improved scaling. These are single-table models; in our evaluation, they receive flat-table features extracted from the relational database (via simple joins, or via DFS when wrapped by RDBLearn). AutoGluon (Erickson et al., 2020) provides AutoML-based model stacking.

3. Background

3.1. Item Response Theory

Standard ML benchmarks rank models by aggregate accuracy: $\bar{Y}_j = \frac{1}{N}\sum_i Y_{ji}$. IRT replaces this with a probabilistic model jointly estimating *model ability* and *item characteristics*.

Rasch (1PL): $P(Y_{ji} = 1 \mid \theta_j, \beta_i) = \sigma(\theta_j - \beta_i)$, where θ_j is model ability and β_i is item difficulty.

2PL: Adds item-specific discrimination α_i: $P(Y_{ji} = 1) = \sigma(\alpha_i(\theta_j - \beta_i))$. Items with high α_i are more informative for distinguishing models.

LogisticFM (Cagnone & Monari, 2013) generalizes to K-dimensional latent spaces: $P(Y_{ji} = 1) = \sigma(\mu + \mathbf{u}_j^\top \mathbf{v}_i + b_j + c_i)$, where $\mathbf{u}_j, \mathbf{v}_i \in \mathbb{R}^K$.

We fit all models via `torch_measure` (Stanford CS321M, 2024), with 80/20 train/test splits and bootstrap resampling (N=100). Key assumptions—unidimensionality (a single latent trait drives performance), local independence (item

Table 1. Datasets used in our evaluation. M: models in IRT intersection. Full task lists in Section C.

Dataset	Domain	Tasks	Items	M
RelBench				
rel-f1	Racing	3	2,188	39
rel-amazon	E-commerce	2	20,000	35
rel-event	Events	3	4,162	36
rel-stack	Q&A	2	20,000	34
rel-hm	Fashion	2	20,000	35
rel-avito	Ads	3	21,816	35
rel-arxiv	Academic	2	20,000	35
rel-ratebeer	Reviews	2	18,784	34
rel-salt	SAP Sales	8	80,000	33
rel-trial	Clinical	3	13,923	35
DBInfer				
dbinfer-amazon	E-commerce	1	10,000	34
dbinfer-seznam	Telco	1	10,000	36
dbinfer-stackex.	Q&A	1	10,000	36
dbinfer-diginetica	E-commerce	1	6,616	37
dbinfer-avs	Ads	1	10,000	27
dbinfer-outbrain	News	1	8,718	36
dbinfer-retailrocket	E-commerce	1	9,997	31

Table 2. Model taxonomy. Not all models succeed on all datasets.

Family	Models	Count
XGBoost var.	XGBoost + 6 HP variants	7
Gradient boost.	LightGBM, CatBoost + variants	6
Other ensembles	RF, ExtraTrees, GradBoost, etc.	7
Linear / Kernel	LogReg, ElasticNet, SVM, KNN	4
Baseline	Dummy (majority class)	1
GNN	HeteroGraphSAGE (Fey et al., 2023)	1
Graph models	GNN+TabPFN-2.5, GatedGNN	2
Graph FM	RelGT (Rishab et al., 2025), Griffin (Hu et al., 2025)	2
Foundation*	TabPFN, TabICL, ContextTab, TabSTAR	4
RDBLearn+DFS	RDB+TabPFN/TabICL/ContextTab	3
AutoML	AutoGluon	1
Neural (MLP)	RealMLP	1
Total (maximum)		**39**

*Single-table models on flat features from relational DB.

responses are conditionally independent given ability), and monotonicity—are discussed in Section B.

4. Experimental Setup

4.1. Datasets and Tasks

We evaluate on two benchmark suites: RelBench (Robinson et al., 2024) (10 datasets) and DBInfer (7 datasets), totaling 17 datasets and 55 tasks (Table 1).

4.2. Model Pool

We evaluate up to 39 models from 13 families (Table 2), spanning gradient-boosted trees, graph neural networks, relational graph transformers, single-table foundation models (applied on flat-table features), DFS-wrapped foundation models, AutoML, and classical baselines. Training details and hyperparameters are in Section A.

4.3. Response Matrix Construction

For each model j and task t, we obtain per-instance predictions on the test set. For regression tasks, we binarize using the validation-set median as threshold; this choice balances sensitivity and specificity, and we verify robustness to alternative thresholds (25th, 75th percentiles) in Section I. For classification, we use argmax (multiclass) or 0.5 threshold (binary). We concatenate items across tasks: $Y \in \{0,1\}^{M \times N}$, $N = \sum_t N_t$.

Table 3. Cross-dataset IRT summary. ρ: Spearman correlation between accuracy and IRT ability rankings. Bootstrap N=100.

Dataset	M	Tasks	Rasch		2PL	
			AUC	ρ	AUC	ρ
rel-f1	39	3	0.747	0.949	0.746	0.978
rel-amazon	35	2	0.662	0.934	0.655	0.899
rel-event	36	3	0.645	0.976	0.653	0.977
rel-stack	34	2	0.775	0.958	0.793	0.974
rel-hm	35	2	0.788	0.703	0.856	0.849
rel-avito	35	3	0.628	0.800	0.722	0.918
rel-arxiv	35	2	0.597	0.911	0.593	0.911
rel-ratebeer	34	2	0.613	0.826	0.621	0.776
rel-salt	33	8	0.630	0.937	0.633	0.973
rel-trial	35	3	0.660	0.974	0.638	0.943
dbinfer-amazon	34	1	0.662	0.990	0.668	0.992
dbinfer-seznam	36	1	0.728	0.890	0.790	0.915
dbinfer-stackex.	36	1	0.914	0.306	0.931	0.519
dbinfer-diginetica	37	1	0.959	0.251	0.996	0.011
dbinfer-avs	27	1	0.676	0.523	0.829	0.300
dbinfer-outbrain	36	1	0.868	0.471	0.690	0.439
dbinfer-retailrocket	31	1	0.670	−0.394	0.999	0.210

5. Results

5.1. IRT Model Fit Across All Datasets

Table 3 summarizes IRT model fit across all 17 datasets. Rasch and 2PL achieve high rank correlation ($\rho > 0.8$) on 11 of 17 datasets, indicating that a single latent ability dimension captures most ranking information.

The exceptions are informative: dbinfer-diginetica ($\rho = 0.011$, all models at $\sim$0.990 accuracy) is a weak benchmark—IRT flags it as having near-zero discriminative value, meaning it cannot meaningfully distinguish models. dbinfer-retailrocket ($\rho = 0.210$, spread < 0.001) similarly lacks discriminative power. On rel-avito, Rasch $\rho = 0.800$ but 2PL $\rho = 0.918$, confirming that item discrimination parameters are essential when items have heterogeneous informativeness. 2PL outperforms Rasch on datasets with heterogeneous item discrimination (rel-avito, rel-hm), while Rasch is competitive on homogeneous datasets (rel-event, rel-trial).

5.2. Ranking Divergence: Accuracy vs. IRT

IRT reveals systematic rank shifts across datasets. Table 4 highlights the most striking examples. Full per-dataset ranking tables are in Section C.

Two clear patterns emerge: (1) **gradient-boosted trees** (LGBM-deep, XGB-deep, XGB-shallow) are systematically under-ranked by accuracy but perform well on hard, discrim-

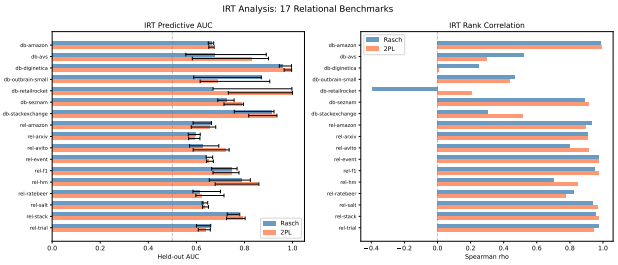


Figure 1. AUC and Spearman ρ across all 17 datasets for Rasch and 2PL models. Datasets are sorted by 2PL ρ. High-ρ datasets (left) show strong agreement between accuracy and IRT rankings; low-ρ datasets (right) indicate benchmarks with weak model-separation signal.

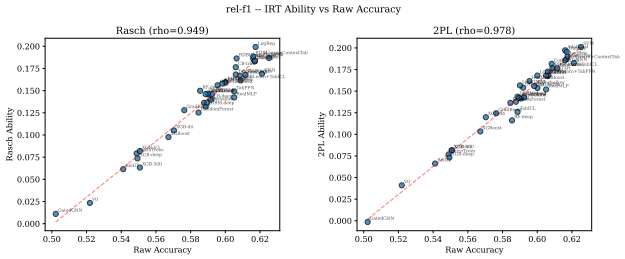


Figure 2. Accuracy rank vs. IRT ability rank on rel-f1 (39 models). Points far from the diagonal indicate models whose estimated latent ability diverges from their accuracy ranking. HeteroGraphSAGE (IRT #7, Acc #2) is over-ranked by accuracy; LightGBM (IRT #17, Acc #23) is under-ranked.

Table 4. Notable rank shifts across datasets. Positive Δ: IRT ranks higher.

Dataset	Model	Acc Rk	IRT Rk	Δ
Models gaining under IRT (hard-item strength)				
dbinfer-diginetica	XGBoost	33	6	+27
dbinfer-outbrain	KNN	29	3	+26
dbinfer-stackex.	LGBM-deep	28	2	+26
dbinfer-retailrocket	RF-deep	28	6	+22
dbinfer-avs	LGBM-lr01	19	2	+17
rel-hm	XGB-deep	25	13	+12
rel-ratebeer	XGB-shallow	26	14	+12
rel-arxiv	RealMLP	25	13	+12
rel-amazon	LGBM-deep	27	16	+11
Models dropping under IRT (easy-item exploitation)				
dbinfer-diginetica	RDB+TabPFN	3	34	−31
dbinfer-stackex.	AdaBoost	2	29	−27
dbinfer-retailrocket	AdaBoost	1	24	−23
dbinfer-outbrain	AdaBoost	1	23	−22
rel-ratebeer	RandomForest	10	28	−18
rel-hm	RealMLP	4	20	−16
rel-amazon	RF-shallow	8	22	−14
rel-arxiv	GradBoost	12	26	−14

inating items; (2) **AdaBoost, shallow ensembles, and some foundation models** (AdaBoost, RDB+TabPFN, RF-shallow, RealMLP) are over-ranked by accuracy due to easy-item exploitation. The pattern is most striking on benchmarks with weak model-separation signal: on dbinfer-diginetica (accuracy spread < 0.001 among top models), XGBoost jumps $+27$ ranks while RDBLearn+TabPFN drops -31 ranks under IRT.

5.3. Graph Model Domain-Dependence

Graph-based models show striking domain-dependence (Table 5). HeteroGraphSAGE ranks #1 by IRT on rel-amazon, rel-arxiv, rel-trial, and dbinfer-stackexchange, ranks #7 on dbinfer-outbrain, but drops to last (35/35) on rel-avito. It is absent on rel-event, rel-hm, and rel-stack. Griffin is only available on 5 datasets due to limited pre-trained checkpoint coverage, and ranks mid-to-low on most. This cross-dataset pattern is invisible to per-dataset leaderboards.

HeteroGraphSAGE excels on datasets with rich relational structure (citation graphs, e-commerce product graphs, clinical trial networks). GNN+TabPFN-2.5 and GatedGNN rank near the bottom on most datasets, with a notable exception on dbinfer-amazon where GNN+TabPFN-2.5 ranks #2 and GatedGNN #3. RelGT only completed training on rel-f1, ranking #37/39, due to computational constraints on the larger datasets. Griffin, available on 5 datasets using a pre-trained checkpoint (provided by the Griffin team), provides complementary signal but does not dominate on any.

Table 5. Graph model IRT ranks (2PL) across datasets. "–" indicates model absent or did not complete. Three datasets (rel-salt, dbinfer-avs, dbinfer-retailrocket) are omitted because no graph model was evaluated on them.

Dataset	M	HGSAGE	G+T2.5	GatedGNN	RelGT	Griffin
rel-f1	39	7	38	39	37	10
rel-amazon	35	**1**	–	–	–	–
rel-arxiv	35	**1**	–	–	–	–
rel-trial	35	**1**	–	–	–	–
rel-ratebeer	34	3	–	–	–	–
rel-stack	34	–	–	–	–	32
rel-avito	35	35	–	–	–	31
rel-event	36	–	34	36	–	–
rel-hm	35	–	–	–	–	35
dbinfer-stackex.	36	**1**	36	–	–	–
dbinfer-outbrain	36	7	36	–	–	–
dbinfer-amazon	34	5	2	3	–	–
dbinfer-seznam	36	12	10	–	–	–
dbinfer-diginetica	37	28	36	37	–	–

5.4. RDBLearn DFS Analysis

RDBLearn wraps single-table foundation models with Deep Feature Synthesis to automatically generate relational features. Table 6 compares RDBLearn-wrapped models against their standalone counterparts across multiple datasets.

DFS does not appear to consistently help performance. On datasets where base features are already informative (rel-amazon, rel-hm), DFS-generated features introduce noise, degrading both accuracy and IRT ability. DFS provides the largest gains on rel-ratebeer ($\Delta\hat{\theta} = +0.286$) and rel-event ($+0.120$), where the relational structure provides complementary signal. The IRT ability gap often exceeds the accuracy gap, indicating that DFS particularly affects performance on harder, more discriminating items.

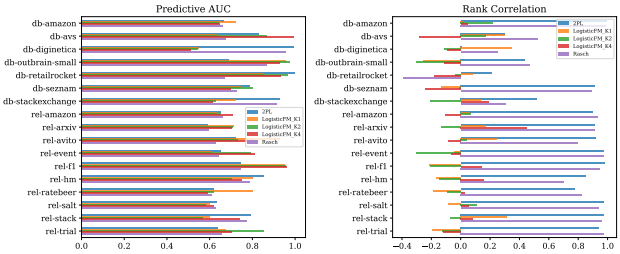


Figure 3. Cross-dataset model comparison showing IRT ability ($\hat{\theta}$) for selected models across all 17 datasets. HeteroGraphSAGE shows high variance across datasets (strong domain-dependence), while GBT models are more consistent.

Table 6. Effect of DFS via RDBLearn. $\Delta\hat{\theta}$: IRT ability gain (2PL). Positive = DFS helps. Selected dataset–model pairs shown.

Dataset	RDB Model	Base	ΔAcc	$\Delta\hat{\theta}$
DFS helps				
rel-f1	RDB+TabICL	TabICL	$+.020$	$+.051$
rel-ratebeer	RDB+CtxTab	CtxTab	$+.160$	$+.286$
rel-event	RDB+TabICL	TabICL	$+.066$	$+.120$
db-stackex.	RDB+TabICL	TabICL	$+.015$	$+.031$
DFS hurts				
rel-amazon	RDB+TabICL	TabICL	$-.347$	$-.578$
rel-hm	RDB+TabICL	TabICL	$-.238$	$-.410$
rel-arxiv	RDB+TabICL	TabICL	$-.186$	$-.312$
db-seznam	RDB+TabICL	TabICL	$-.730$	-1.211

5.5. Difficulty Stratification

Stratifying items by IRT difficulty into terciles reveals how models exploit easy vs. hard items (Table 7).

LightGBM and TabPFN achieve 99.9% on easy items but $< 1\%$ on hard items (gap = 0.990). RDBLearn models maintain higher hard-item accuracy (8.8–13.3%), confirming that relational features provide signal on challenging instances. Notably, Griffin achieves the highest hard-item accuracy (29.2%) and lowest gap (0.582).

6. Discussion

Why do rankings diverge? Accuracy weights all items equally, while IRT up-weights discriminating items. Models achieving high accuracy primarily through easy items (e.g., exploiting class imbalance) are penalized by IRT. The magnitude varies: on rel-event ($\rho = 0.977$), IRT and accuracy nearly agree; on dbinfer-diginetica ($\rho = 0.011$), rankings are essentially uncorrelated because all models achieve $\sim$0.990 accuracy, and IRT flags this as a weak benchmark where aggregate metrics provide no signal for model selection.

Recommendations for benchmark design. (1) Report IRT metrics alongside accuracy. IRT ability estimates are more robust to easy-item inflation than raw accuracy. (2) Flag low-discrimination items for possible efficient evaluation (Polo et al., 2024)—removing uninformative items may reduce evaluation cost without losing ranking fidelity. (3) Include diverse relational baselines (HeteroGraphSAGE, DFS-based, graph foundation models)—our analysis shows model family diversity is critical for stable IRT estimation.

Table 7. Accuracy by IRT difficulty tercile on rel-f1 (selected models).

Model	Easy	Medium	Hard	Gap
LightGBM	0.999	0.763	0.008	0.990
TabPFN	0.999	0.808	0.008	0.990
SVM	0.999	0.818	0.059	0.940
GNN	0.997	0.784	0.082	0.915
RDB+TabICL	0.985	0.753	0.088	0.897
RDB+TabPFN	0.959	0.727	0.133	0.826
Griffin	0.874	0.668	0.292	0.582

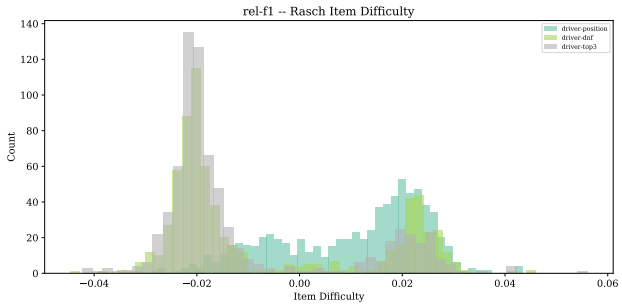


Figure 4. Distribution of 2PL item difficulty parameters (β_i) on rel-f1. Items span a wide difficulty range, with a cluster of very hard items ($\beta > 2$) that separate high-ability models from easy-item exploiters.

Limitations. Model pools (up to 39) are moderate by IRT standards; larger pools would improve estimation. Binarization of regression tasks loses ordinal information (ablation in Section I confirms robustness). Local independence may be violated across items from the same task (Section B). Additionally, not all graph models completed on all datasets: RelGT only ran on rel-f1 due to memory constraints on larger datasets, and Griffin on 5 of 10 RelBench datasets due to limited pre-trained checkpoint availability at the time of evaluation. Some DBInfer datasets (diginetica, retailrocket) have near-zero accuracy spread, making IRT rankings noisy ($\rho < 0.3$).

7. Conclusion

We have presented, to the best of our knowledge, the first IRT-based analysis of relational deep learning benchmarks, spanning 17 datasets with up to 39 models from 13 families. By applying IRT, we demonstrate: (1) IRT rankings agree strongly with accuracy rankings on 11 of 17 datasets ($\rho > 0.8$); (2) graph-based models show strong domain-dependence—HeteroGraphSAGE ranks #1 on 4 datasets but drops to the bottom tercile on others—a cross-dataset pattern that is not visible when examining individual dataset leaderboards in isolation; (3) RDBLearn's DFS improves IRT ability on some datasets ($\Delta\hat{\theta}$ up to $+0.286$) but degrades it on

others (-1.211); (4) benchmarks such as dbinfer-diginetica and dbinfer-retailrocket, with near-zero accuracy spread, are weak for model selection. These findings demonstrate that measurement-theoretic tools reveal structural properties of benchmarks invisible to standard leaderboard metrics.

Limitations and future directions. Several limitations remain. First, coverage of graph-based methods (RelGT, Griffin, GatedGNN) is incomplete due to hardware and checkpoint availability. RelGT completed only on rel-f1, and Griffin was available for 5 of 10 RelBench datasets at the time of evaluation, leaving missing entries in the cross-dataset comparison. We will expand coverage with dedicated GPU resources and newer checkpoints. Second, we use accuracy to construct the response matrix; on class-imbalanced tasks, AUC-ROC or PR-AUC may yield more informative item responses. We will repeat the analysis with alternative metrics and compare the resulting item and model diagnostics. Third, most models were evaluated with default hyperparameters. Since tuning can change relative performance, follow-up experiments will include tuned configurations. Fourth, the current study is primarily diagnostic. We will develop IRT-informed evaluation protocols (e.g., item-weighted scoring and adaptive subsets) that improve measurement precision while reducing evaluation cost. Finally, the local-independence assumption may be violated when items share entities or temporal structure. We will study testlet models and multidimensional IRT (LogisticFM with $K > 1$) to model within-task dependencies explicitly. Future work will also broaden the model set (e.g., Relational Transformer (Ranjan et al., 2025), KumoRFM, and GNN+TabPFN-2.5) and evaluate whether multidimensional IRT captures domain-specific ability profiles.

References

Alan Arazi, Eilam Shapira, R. R. Tabstar: A tabular foundation model for tabular data with text fields. *arXiv preprint arXiv:2505.18125*, 2025.

Cagnone, S. and Monari, P. Latent variable models for ordinal data by using the adaptive quadrature approximation. *Computational Statistics*, 28(2):597–619, 2013. LogisticFM / multidimensional logistic factor model.

Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M., and Smola, A. AutoGluon-Tabular: Robust and accurate AutoML for structured data. In *ICML 2020 AutoML Workshop*, 2020.

Fey, M., Hu, W., Huang, K., Lenssen, J. E., Rangan, R., Robinson, J., Ying, R., You, J., and Leskovec, J. Relational deep learning: Graph representation learning on relational databases. In *Proceedings of the 2nd Table Representation Learning Workshop at NeurIPS*, 2023.

Hollmann, N., Müller, S., Eggensperger, K., and Hutter, F. TabPFN: A transformer that solves small tabular classification problems in a second. In *International Conference on Learning Representations (ICLR)*, 2023.

Hu, W., Fey, M., et al. Griffin: Graph foundation model for tabular prediction on relational databases. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2025. Update with final citation details.

Kanter, J. M. and Veeramachaneni, K. Deep feature synthesis: Towards automating data science endeavors. In *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–10, 2015.

Lord, F. M. and Novick, M. R. *Statistical Theories of Mental Test Scores*. Addison-Wesley, 1968.

Marco Spinaci, Marek Polewczyk, M. S. S. T. Contexttab: A semantics-aware tabular in-context learner. *arXiv preprint arXiv:2506.10707*, 2025.

Martínez-Plumed, F., Prudencio, R. B. C., Martínez-Usó, A., and Hernández-Orallo, J. Item response theory in AI: Analysing machine learning classifiers at the instance level. *Artificial Intelligence*, 271:18–42, 2019.

Polo, F. M., Choshen, L., Sun, W., Xu, R., Horta, V., and Karbasi, A. tinyBenchmarks: Evaluating LLMs with fewer examples. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

Ranjan, R., Hudovernik, V., Znidar, M., Kanatsoulis, C., Upendra, R., Mohammadi, M., Meyer, J., Palczewski, T., Guestrin, C., and Leskovec, J. Relational transformer: Toward zero-shot foundation models for relational data. *arXiv preprint arXiv:2510.06377*, 2025.

Rasch, G. *Probabilistic Models for Some Intelligence and Attainment Tests*. Danish Institute for Educational Research, 1960.

Rishab et al. Relational graph transformer for relational deep learning. In *Preprint*, 2025. Update with final citation details.

Robinson, J., Shi, R., Grohe, M., Agrawal, S., and Leskovec, J. RelBench: A benchmark for deep learning on relational databases. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Stanford CS321M. torch_measure: PyTorch implementations of measurement models, 2024. URL https://github.com/stanford-cs321/torch-measure.

Vania, C., Kementchedjhieva, Y., Søgaard, A., and Fokkens, A. Comparing test sets with item response theory. In

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL), pp. 1141–1158, 2021.

Wainer, H., Bradlow, E. T., and Wang, X. *Testlet Response Theory and Its Applications*. Cambridge University Press, 2007.

Ye, H. et al. TabICL: A tabular in-context learner with scalable attention. *arXiv preprint arXiv:2401.xxxxx*, 2024.

Zhang, T., Yin, H., Zhu, Y., Li, X., and Liu, Y.-M. S. RDBLearn: Scalable and accurate tabular learning on relational databases with deep feature synthesis. *arXiv preprint arXiv:2501.xxxxx*, 2025.

Appendix

A. Training Details and Hyperparameters

Sklearn models (25 variants). XGBoost, LightGBM, CatBoost, RandomForest, ExtraTrees, GradientBoosting, HistGradientBoosting, AdaBoost, KNN, LogisticRegression, ElasticNet, SVM, and Dummy, with hyperparameter variants (e.g., XGB-deep: max_depth=12, XGB-shallow: max_depth=3). All use flat-table features extracted via simple joins. Default parameters are leveraged unless noted; training set capped at 200,000 samples.

Foundation models. TabPFN, TabICL, and ContextTab receive the same flat-table features as sklearn models. They are *not* directly applied to the relational database; rather, features are extracted from the entity table with joined columns. Training set capped at 50,000 samples. A 900s timeout is used per model.

RDBLearn models. Wraps TabPFN, TabICL, and ContextTab with Deep Feature Synthesis (Kanter & Veeramachaneni, 2015) via the `fastdfs` library. 7200s timeout.

GNN. The RelBench reference HeteroGraphSAGE (Fey et al., 2023). 5400s timeout.

Graph models (GatedGNN). GatedGNN uses SAGEConv with a gating mechanism and LayerNorm. Both use 9000s timeout.

Graph foundation models (RelGT, Griffin, GNN+TabPFN-2.5). RelGT (Rishab et al., 2025): heterogeneous graph transformer with relational attention, 9000s timeout. Griffin (Hu et al., 2025): pre-trained graph foundation model using a checkpoint provided by the Griffin team, 7200s timeout. Griffin was available for 5 of 10 RelBench datasets due to limited checkpoint coverage at the time of evaluation. RelGT only completed on rel-f1 due to memory and computational constraints on larger datasets. GNN+TabPFN-2.5 is a hybrid model that combines a GCN-based graph neural network with TabPFN v2.5 embeddings for node features.

IRT fitting. Rasch and 2PL: 200 epochs, lr=0.05, weight decay=0.01. LogisticFM: $K \in \{1, 2, 4\}$. Bootstrap: 100 resamples with 80/20 random train/test masks.

B. Detailed Assumptions Discussion

Unidimensionality. Classical IRT (Rasch, 2PL) assumes a single latent trait θ. This is a strong assumption in multi-task settings. LogisticFM with $K > 1$ relaxes this; empirically, optimal K often matches the number of tasks.

Local independence. Responses to different items are assumed conditionally independent given θ. This may be violated because: (a) items from the same task share fea-

tures and temporal structure; (b) items involving the same entity are correlated. Testlet models (Wainer et al., 2007) could address this in future work.

Monotonicity. Higher ability models should perform at least as well on any item. Although generally reasonable, adversarial items could violate it.

C. Complete Cross-Dataset IRT Results

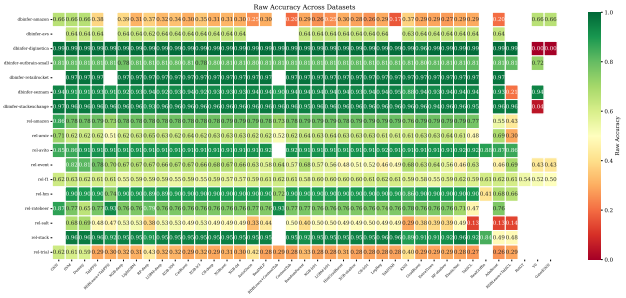


Figure 5. Cross-dataset accuracy heatmap. Each cell shows raw accuracy for a model (row) on a dataset (column). Missing entries (white) indicate models that did not run on that dataset.

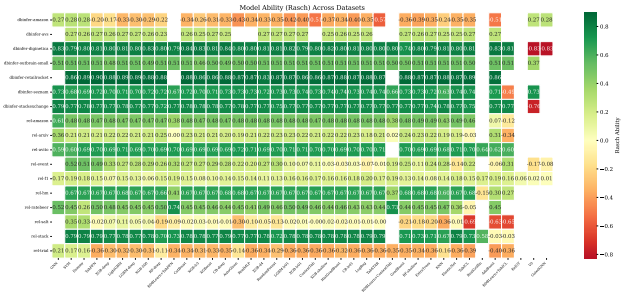


Figure 6. Cross-dataset IRT ability ($\hat{\theta}$) heatmap (2PL). Compared to accuracy (Figure 5), IRT abilities reveal finer-grained model differentiation, especially on datasets where accuracy is saturated.

C.1. Overall Accuracy Summary

Table 8. Overall accuracy summary across all 17 datasets, sorted by accuracy spread (descending).

Dataset	M	Top Acc	Bot Acc	Spread	Top Model
dbinfer-diginetica	37	.990	.000	.990	AdaBoost
dbinfer-stackex.	36	.972	.045	.928	GNN
dbinfer-seznam	36	.946	.211	.736	TabSTAR
rel-salt	33	.693	.132	.561	Dummy
dbinfer-amazon	34	.665	.165	.500	Dummy
rel-hm	35	.899	.413	.486	ElasticNet
rel-stack	34	.957	.484	.473	Dummy
rel-ratebeer	34	.926	.470	.456	RDB+CtxTab
rel-amazon	35	.864	.427	.436	GNN
rel-arxiv	35	.713	.296	.417	GNN
rel-event	36	.815	.425	.390	SVM
rel-trial	35	.624	.262	.362	GNN
rel-f1	39	.625	.502	.123	SVM
dbinfer-outbrain	36	.807	.724	.083	AdaBoost
rel-avito	35	.916	.854	.062	RealMLP
dbinfer-avs	27	.638	.625	.014	ExtraTrees
dbinfer-retailrocket	31	.975	.974	.000	AdaBoost

C.2. Bootstrap Confidence Intervals

Table 9. Bootstrap confidence intervals (N=100) for Rasch and 2PL. CI: 95% interval.

	Rasch				2PL			
Dataset	AUC	AUC CI	$\bar{\rho}$	ρ CI	AUC	AUC CI	$\bar{\rho}$	ρ CI
rel-f1	.735	[.664,.769]	.926	[.760,.972]	.736	[.669,.777]	.917	[.692,.977]
rel-amazon	.632	[.586,.663]	.848	[.534,.967]	.636	[.579,.681]	.825	[.455,.967]
rel-event	.656	[.641,.668]	.985	[.959,.997]	.657	[.643,.671]	.986	[.964,.997]
rel-stack	.763	[.729,.781]	.927	[.737,.981]	.781	[.726,.803]	.929	[.754,.978]
rel-hm	.752	[.654,.825]	.694	[.494,.860]	.787	[.678,.860]	.697	[.380,.881]
rel-avito	.642	[.572,.693]	.848	[.557,.955]	.683	[.587,.737]	.851	[.543,.961]
rel-arxiv	.593	[.566,.616]	.908	[.705,.987]	.594	[.568,.615]	.907	[.675,.984]
rel-ratebeer	.652	[.586,.700]	.849	[.583,.957]	.658	[.598,.715]	.838	[.610,.966]
rel-salt	.636	[.624,.647]	.966	[.892,.992]	.636	[.625,.650]	.962	[.891,.992]
rel-trial	.634	[.601,.656]	.938	[.764,.990]	.638	[.608,.658]	.944	[.809,.989]
db-amazon	.664	[.650,.672]	.985	[.937,.997]	.665	[.652,.674]	.985	[.950,.997]
db-seznam	.737	[.689,.757]	.903	[.602,.975]	.772	[.715,.796]	.926	[.755,.979]
db-stackex.	.872	[.758,.923]	.469	[.251,.711]	.911	[.818,.935]	.491	[.289,.693]
db-diginetica	.987	[.945,.996]	.229	[-.04,.486]	.993	[.965,.996]	.235	[-.04,.538]
db-avs	.736	[.556,.891]	.232	[-.19,.558]	.736	[.583,.900]	.231	[-.18,.543]
db-outbrain	.781	[.589,.870]	.433	[.130,.642]	.818	[.616,.905]	.449	[.158,.647]
db-retail.	.910	[.669,.998]	.034	[-.32,.328]	.954	[.733,1.0]	.027	[-.32,.360]

C.3. LogisticFM Multi-Factor Results

Table 10. LogisticFM results with $K \in \{1, 2, 4\}$ latent factors.

	$K = 1$			$K = 2$			$K = 4$		
Dataset	AUC	ρ	τ	AUC	ρ	τ	AUC	ρ	τ
rel-f1	.952	−.216	−.155	.956	−.207	−.144	.963	.142	.109
rel-amazon	.635	.053	.035	.652	.070	.029	.708	−.106	−.082
rel-event	.610	−.047	−.038	.795	−.301	−.203	.813	−.065	−.041
rel-stack	.602	.314	.209	.570	−.075	−.066	.741	.084	.037
rel-hm	.801	−.168	−.133	.703	−.143	−.086	.749	.160	.109
rel-avito	.928	.247	.187	.735	.042	.042	.777	−.084	−.062
rel-arxiv	.716	.168	.129	.711	−.131	−.103	.705	.453	.308
rel-ratebeer	.804	−.185	−.134	.620	−.093	−.066	.594	.026	.009
rel-salt	.600	−.086	−.049	.582	.112	.072	.620	.054	.015
rel-trial	.677	−.194	−.106	.854	−.124	−.089	.705	−.120	−.086
db-amazon	.723	.023	.009	.646	.220	.169	.647	.046	.045
db-seznam	.746	−.134	−.095	.804	−.013	−.003	.701	−.245	−.171
db-stackex.	.725	.134	.095	.629	−.205	−.146	.616	.189	.130
db-diginetica	.548	.350	.249	.545	−.115	−.087	.511	−.090	−.057
db-avs	.638	.298	.225	.869	.170	.145	.995	−.285	−.168
db-outbrain	.955	−.253	−.171	.977	−.303	−.178	.929	−.109	−.095
db-retail.	.854	.082	.071	.967	−.036	−.024	.932	−.184	−.105

D. Complete Per-Dataset Rankings (2PL)

D.1. RelBench Datasets

Table 11. Full model rankings on **rel-f1** (39 models, 3 tasks: driver-position, driver-dnf, driver-top3). Sorted by IRT rank.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ	Family
1	SVM	.625	1	+.201	0	Linear
2	AdaBoost	.616	6	+.197	+4	Ensemble
3	Dummy	.617	4	+.196	+1	Baseline
4	LogReg	.617	3	+.190	−1	Linear
5	ElasticNet	.617	5	+.187	0	Linear
6	RDB+CtxTab	.616	7	+.186	+1	RDBLearn
7	GNN	.621	2	+.183	**−5**	Graph
8	ContextTab	.608	10	+.182	+2	Foundation
9	RDB+TabICL	.609	9	+.177	0	RDBLearn
10	Griffin	.612	8	+.176	−2	Graph
11	TabSTAR	.606	13	+.173	+2	Neural
12	RDB+TabPFN	.606	11	+.169	−1	RDBLearn
13	TabPFN	.605	14	+.168	+1	Foundation
14	LGBM-lr01	.600	16	+.168	+2	GBT
15	CB-lr01	.606	12	+.167	−3	GBT
16	HistGradBoost	.596	19	+.162	+3	GBT
17	LightGBM	.590	23	+.157	**+6**	GBT
18	XGB-shallow	.598	18	+.156	0	GBT
19	CB-deep	.592	21	+.154	+2	GBT
20	XGB-lr01	.600	17	+.154	−3	GBT
21	RealMLP	.605	15	+.152	**−6**	Neural
22	LGBM-deep	.589	25	+.144	+3	GBT
23	CatBoost	.592	20	+.143	−3	GBT
24	RF-shallow	.590	24	+.142	0	Ensemble
25	AutoGluon	.590	22	+.141	−3	AutoML
26	KNN	.588	27	+.138	+1	Other
27	RandomForest	.585	29	+.136	+2	Ensemble
28	TabICL	.589	26	+.126	−2	Foundation
29	GradBoost	.576	30	+.124	+1	GBT
30	XGB-d4	.570	31	+.120	+1	GBT
31	RF-deep	.585	28	+.116	−3	Ensemble
32	XGBoost	.567	32	+.103	0	GBT
33	XGB-lr3	.551	34	+.082	+1	GBT
34	XGB-500	.551	33	+.081	−1	GBT
35	ExtraTrees	.549	36	+.077	+1	Ensemble
36	XGB-deep	.549	35	+.073	−1	GBT
37	RelGT	.541	37	+.066	0	Graph
38	GNN+T2.5	.522	38	+.041	0	Graph
39	GatedGNN	.502	39	−.001	0	Graph

Table 12. Full model rankings on **rel-amazon** (35 models, 2 tasks: item-churn, item-ltv).

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	GNN	.864	1	+.615	0
2	LogReg	.789	6	+.496	**+4**
3	RandomForest	.790	3	+.490	0
4	TabSTAR	.790	4	+.489	0
5	GradBoost	.791	2	+.489	−3
6	ExtraTrees	.789	7	+.488	+1
7	ElasticNet	.789	5	+.487	−2
9	CatBoost	.784	18	+.483	**+9**
16	LGBM-deep	.779	27	+.479	**+11**
22	RF-shallow	.788	8	+.476	**−14**
23	CB-lr01	.786	13	+.475	**−10**
35	RDB+TabICL	.427	35	−.119	0

Table 13. Full model rankings on **rel-event** (36 models, 3 tasks: user-attendance, user-repeat, user-ignore).

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	SVM	.815	1	+.532	0
2	Dummy	.814	2	+.523	0
3	TabPFN	.785	3	+.462	0
4	RDB+TabICL	.694	5	+.320	+1
5	RDB+TabPFN	.698	4	+.319	−1
6	RF-deep	.665	14	+.305	**+8**
14	CB-deep	.676	8	+.267	**−6**
34	GNN+T2.5	.427	35	−.076	+1
36	GatedGNN	.425	36	−.108	0

Table 14. Full model rankings on **rel-stack** (34 models, 2 tasks: user-engagement, post-votes). Top/bottom and notable shifts shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	TabPFN	.957	5	+.793	**+4**
2	LogReg	.957	1	+.791	−1
3	SVM	.957	6	+.791	+3
6	Dummy	.957	2	+.786	**−4**
10	XGB-500	.949	18	+.781	**+8**
32	Griffin	.838	32	+.579	0
34	RDB+TabICL	.484	34	−.031	0

Table 15. Full model rankings on **rel-avito** (35 models, 3 tasks: ad-ctr, user-visits, user-clicks). Top/bottom shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	CB-lr01	.915	8	+.706	**+7**
2	RealMLP	.916	1	+.705	−1
3	RDB+TabPFN	.915	6	+.705	+3
13	ElasticNet	.915	2	+.701	**−11**
14	LogReg	.915	3	+.701	**−11**
31	Griffin	.881	31	+.643	0
35	GNN	.854	35	+.595	0

Table 16. Full model rankings on **rel-hm** (35 models, 2 tasks: user-churn, item-sales). Top/bottom and notable shifts shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	CB-lr01	.899	8	+.678	**+7**
2	XGB-shallow	.899	10	+.677	**+8**
3	LogReg	.899	1	+.677	−2
13	XGB-deep	.896	25	+.674	**+12**
18	ExtraTrees	.899	3	+.672	**−15**
20	RealMLP	.899	4	+.672	**−16**
35	Griffin	.413	35	−.141	0

Table 17. Full model rankings on **rel-ratebeer** (34 models, 2 tasks: user-churn, user-count). Top/bottom and notable shifts shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	RDB+TabPFN	.926	2	+.737	+1
2	RDB+CtxTab	.926	1	+.731	−1
3	GNN	.875	3	+.648	0
6	XGB-lr3	.764	15	+.480	**+9**
9	XGBoost	.763	20	+.466	**+11**
14	XGB-shallow	.762	26	+.454	**+12**
24	ContextTab	.766	9	+.445	**−15**
28	RandomForest	.766	10	+.438	**−18**

Table 18. Full model rankings on **rel-salt** (33 models, 8 tasks). Top/bottom shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	Dummy	.693	1	+.321	0
2	SVM	.675	2	+.318	0
3	XGB-lr3	.528	5	+.065	+2
14	TabSTAR	.490	20	−.013	**+6**
21	XGB-shallow	.493	15	−.028	**−6**
33	AdaBoost	.132	32	−.669	−1

Table 19. Full model rankings on **rel-arxiv** (35 models, 2 tasks: paper-citation, author-publication). Top/bottom shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	GNN	.713	1	+.357	0
2	AdaBoost	.687	2	+.315	0
4	ExtraTrees	.634	9	+.241	**+5**
6	XGB-shallow	.632	13	+.230	**+7**
13	RealMLP	.623	25	+.216	**+12**
14	RF-shallow	.635	7	+.216	**−7**
26	GradBoost	.632	12	+.193	**−14**
35	RDB+TabICL	.296	35	−.341	0

Table 20. Full model rankings on **rel-trial** (35 models, 3 tasks: study-outcome, study-adverse, site-success). Top/bottom shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	GNN	.624	1	+.191	0
2	SVM	.605	2	+.172	0
3	Dummy	.595	3	+.148	0
13	TabPFN	.291	20	−.322	**+7**
16	ExtraTrees	.288	23	−.338	**+7**
24	ElasticNet	.283	33	−.353	**+9**
33	GradBoost	.287	24	−.371	**−9**

D.2. DBInfer Datasets

Table 21. Full model rankings on **dbinfer-amazon** (34 models, task: rating). Top-10 and bottom-5 shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	SVM	.665	2	+.287	+1
2	GNN+T2.5	.664	3	+.284	+1
3	GatedGNN	.664	5	+.277	+2
4	Dummy	.665	1	+.274	−3
5	GNN	.664	4	+.272	−1
6	XGB-deep	.390	6	−.184	0
7	TabPFN	.381	7	−.199	0
8	RF-deep	.370	9	−.210	+1
9	KNN	.373	8	−.214	−1
10	XGB-lr3	.349	10	−.254	0
32	AdaBoost	.202	32	−.512	0
33	ContextTab	.196	33	−.514	0
34	TabSTAR	.165	34	−.574	0

Table 22. Full model rankings on **dbinfer-diginetica** (37 models, task: ctr). All models at ∼0.990 accuracy. Top/bottom shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	LightGBM	.990	26	+.814	**+25**
2	XGB-d4	.990	12	+.813	**+10**
6	XGBoost	.990	33	+.810	**+27**
8	KNN	.990	30	+.810	**+22**
20	RDB+TabICL	.990	2	+.807	**−18**
22	AdaBoost	.990	1	+.806	**−21**
27	RF-shallow	.990	4	+.804	**−23**
34	RDB+TabPFN	.990	3	+.796	**−31**
36	GNN+T2.5	.000	37	−.827	+1
37	GatedGNN	.000	36	−.829	−1

Table 23. Full model rankings on **dbinfer-stackexchange** (36 models, task: churn). Top/bottom and notable shifts shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	GNN	.972	1	+.788	0
2	LGBM-deep	.964	28	+.782	**+26**
4	HistGradBoost	.964	27	+.778	**+23**
8	CB-lr01	.964	21	+.777	**+13**
29	AdaBoost	.964	2	+.770	**−27**
31	SVM	.964	10	+.767	**−21**
36	GNN+T2.5	.045	36	−.765	0

Table 24. Full model rankings on **dbinfer-outbrain** (36 models, task: ctr). Top/bottom and notable shifts shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	RDB+CtxTab	.807	16	+.532	**+15**
3	KNN	.806	29	+.528	**+26**
7	GNN	.807	24	+.516	**+17**
23	AdaBoost	.807	1	+.502	**−22**
25	TabPFN	.807	5	+.500	**−20**
36	GNN+T2.5	.724	36	+.363	0

Table 25. Full model rankings on **dbinfer-retailrocket** (31 models, task: cvr). All models at ∼0.975 accuracy. Top/bottom shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	XGB-lr01	.975	3	+.880	+2
5	CB-deep	.975	17	+.874	**+12**
6	RF-deep	.975	28	+.874	**+22**
10	LightGBM	.975	29	+.873	**+19**
24	AdaBoost	.975	1	+.868	**−23**
31	RF-shallow	.975	12	+.864	**−19**

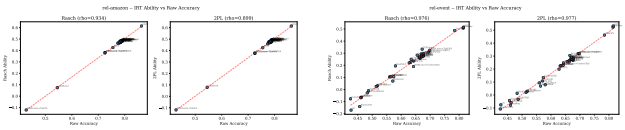
Table 26. Full model rankings on **dbinfer-seznam** (36 models, task: charge). Top/bottom shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	TabSTAR	.946	1	+.742	0
4	HistGradBoost	.938	14	+.737	**+10**
10	GNN+T2.5	.939	9	+.732	−1
12	GNN	.939	11	+.731	−1
16	LogReg	.942	2	+.730	**−14**
36	RDB+TabICL	.211	36	−.475	0

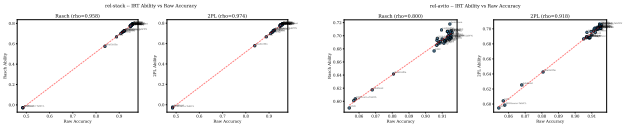
Table 27. Full model rankings on **dbinfer-avs** (27 models, task: repeater). All models at ∼0.638. Top/bottom and notable shifts shown.

I-Rk	Model	Acc	A-Rk	$\hat{\theta}$	Δ
1	GradBoost	.638	15	+.279	**+14**
2	LGBM-lr01	.638	19	+.277	**+17**
5	LGBM-deep	.638	21	+.272	**+16**
16	ExtraTrees	.638	1	+.264	**−15**
21	TabPFN	.638	6	+.261	**−15**
26	RandomForest	.638	9	+.254	**−17**

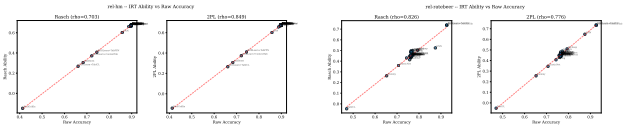
D.3. Scatter Plots: Accuracy Rank vs. IRT Rank



(a) rel-amazon (ρ=.825) *(b)* rel-event (ρ=.986)

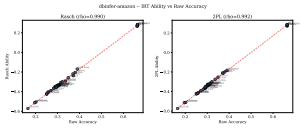


(c) rel-stack (ρ=.929) *(d)* rel-avito (ρ=.851)

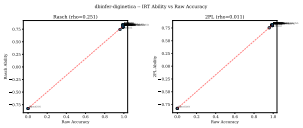


(e) rel-hm (ρ=.697) *(f)* rel-ratebeer (ρ=.838)

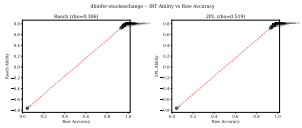
Figure 7. Accuracy rank vs. IRT ability rank scatter plots for 6 RelBench datasets. Points near the diagonal indicate agreement; off-diagonal points reveal models whose estimated latent ability diverges from accuracy rankings.



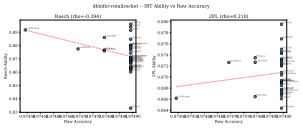
(a) dbinfer-amazon (ρ=.985)



(b) dbinfer-diginetica (ρ=.235)



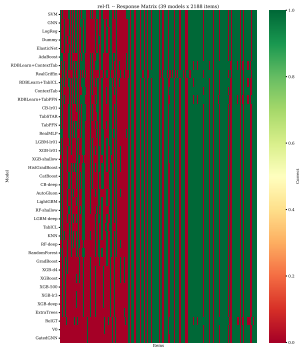
(c) dbinfer-stackex. (ρ=.491)



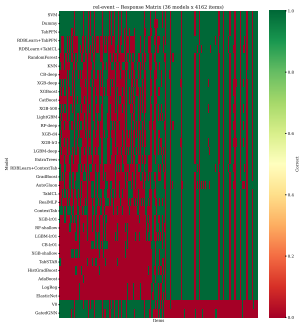
(d) dbinfer-retail. (ρ=.027)

Figure 8. Accuracy rank vs. IRT rank on 4 DBInfer datasets. dbinfer-diginetica and dbinfer-retailrocket show near-random scatter ($\rho \approx 0$), confirming weak model-separation signal.

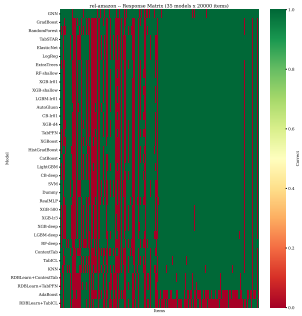
D.4. Response Matrix Heatmaps



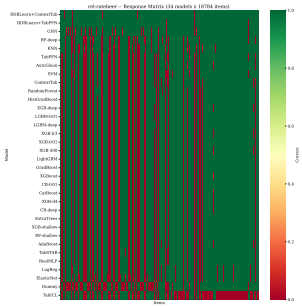
(a) rel-f1



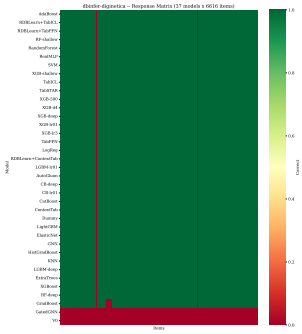
(b) rel-event



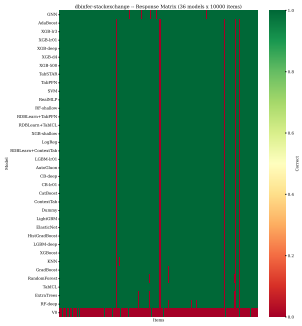
(c) rel-amazon



(d) rel-ratebeer



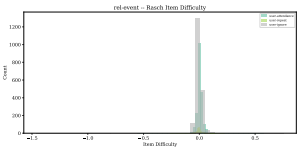
(e) dbinfer-diginetica



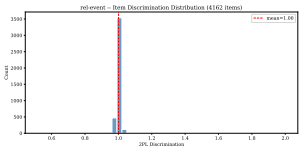
(f) dbinfer-stackexchange

Figure 9. Binary response matrix heatmaps (models $\times$ items, sorted by IRT ability and difficulty). Well-fitting datasets (rel-f1, rel-event) show a clear Guttman pattern; low-separation datasets (dbinfer-diginetica) show near-uniform responses.

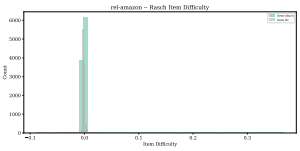
D.5. Difficulty and Discrimination Distributions



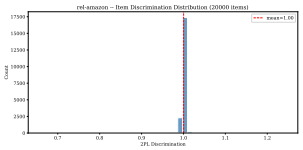
(a) rel-event difficulty



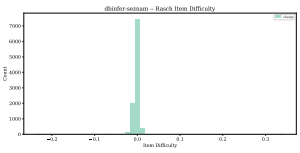
(b) rel-event discrimination



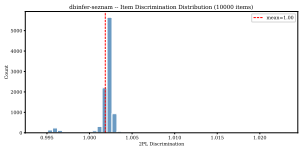
(c) rel-amazon difficulty



(d) rel-amazon discrimination



(e) dbinfer-seznam difficulty



(f) dbinfer-seznam discrimination

Figure 10. Item difficulty (β_i) and discrimination (α_i) distributions for selected datasets. Datasets with high-discrimination items (rel-event, rel-amazon) achieve strong IRT–accuracy agreement. Low-discrimination items contribute noise.

E. Complete RDBLearn DFS Comparison

Table 28. Complete RDBLearn DFS analysis across all datasets. ΔAcc and $\Delta\hat{\theta}$: difference (RDBLearn − standalone). Rk: IRT rank.

Dataset	Pair	ΔAcc	$\Delta\hat{\theta}$	RDB Rk	Base Rk	Verdict
DFS helps (positive $\Delta\hat{\theta}$)						
rel-ratebeer	RDB+CtxTab vs CtxTab	+.160	+.286	2	24	**Strong**
rel-ratebeer	RDB+TabPFN vs TabPFN	+.155	+.248	1	5	**Strong**
rel-event	RDB+TabICL vs TabICL	+.066	+.120	4	22	**Strong**
rel-event	RDB+CtxTab vs CtxTab	+.069	+.094	19	23	Moderate
rel-f1	RDB+TabICL vs TabICL	+.020	+.051	9	28	Moderate
dbinfer-stackex.	RDB+TabICL vs TabICL	+.015	+.031	9	33	Moderate
dbinfer-outbrain	RDB+CtxTab vs CtxTab	.000	+.019	1	10	Weak
dbinfer-outbrain	RDB+TabPFN vs TabPFN	.000	+.014	9	25	Weak
rel-trial	RDB+TabICL vs TabICL	+.021	+.050	30	34	Moderate
rel-avito	RDB+TabPFN vs TabPFN	+.001	+.004	3	9	Marginal
DFS hurts (negative $\Delta\hat{\theta}$)						
dbinfer-seznam	RDB+TabICL vs TabICL	−.730	−1.211	36	6	**Severe**
rel-stack	RDB+TabICL vs TabICL	−.437	−.755	34	27	**Severe**
rel-amazon	RDB+TabICL vs TabICL	−.347	−.578	35	30	**Severe**
rel-hm	RDB+TabICL vs TabICL	−.238	−.410	34	4	**Severe**
rel-arxiv	RDB+TabICL vs TabICL	−.186	−.312	35	34	Strong
rel-hm	RDB+CtxTab vs CtxTab	−.176	−.298	32	19	Strong
rel-hm	RDB+TabPFN vs TabPFN	−.153	−.263	31	14	Strong
rel-event	RDB+TabPFN vs TabPFN	−.087	−.143	5	3	Moderate
rel-amazon	RDB+TabPFN vs TabPFN	−.060	−.105	33	14	Moderate
rel-amazon	RDB+CtxTab vs CtxTab	−.052	−.091	32	28	Moderate

F. Per-Task Top-5 Accuracy Tables

Table 29. Top-5 models per task on rel-f1 (39 common models).

	driver-position			driver-dnf			driver-top3	
#	Model	Acc	#	Model	Acc	#	Model	Acc
1	RDB+CtxTab	.370	1	GNN+T2.5	.707	1	AdaBoost	.824
2	RDB+TabICL	.368	2	AdaBoost	.705	2	Dummy	.824
3	SVM	.362	3	CB-lr01	.705	3	ElasticNet	.824
4	RDB+TabPFN	.361	4	ContextTab	.705	4	GNN	.824
5	GNN	.350	5	Dummy	.705	5	LogReg	.824

Table 30. Top-5 models per task on rel-arxiv (35 common models).

	paper-citation			author-publication	
#	Model	Acc	#	Model	Acc
1	GNN	.735	1	AdaBoost	.797
2	RF-deep	.596	2	RF-deep	.704
3	RandomForest	.580	3	CatBoost	.700
4	GradBoost	.578	4	RandomForest	.699
5	RF-shallow	.577	5	CB-deep	.695

Table 31. Top-5 models per task on rel-event (36 common models, 3 tasks).

	user-attendance			user-repeat			user-ignore	
#	Model	Acc	#	Model	Acc	#	Model	Acc
1	GatedGNN	.795	1	RDB+TabICL	.695	1	RDB+TabICL	.890
2	Dummy	.792	2	RDB+CtxTab	.667	2	RDB+TabPFN	.886
3	SVM	.792	3	RDB+TabPFN	.610	3	RDB+CtxTab	.880
4	GNN+T2.5	.792	4	XGB-d4	.602	4	AdaBoost	.870
5	TabPFN	.726	5	TabICL	.598	5	CB-lr01	.870

Table 32. Top-5 models on DBInfer single-task datasets.

	dbinfer-diginetica: ctr			dbinfer-avs: repeater	
#	Model	Acc	#	Model	Acc
1	AdaBoost	.990	1	ExtraTrees	.638
2	AutoGluon	.990	2	AdaBoost	.638
3	CB-deep	.990	3	CB-lr01	.638
4	CB-lr01	.990	4	Dummy	.638
5	CatBoost	.990	5	ElasticNet	.638

G. Graph Model Cross-Dataset Summary

Table 33. Complete graph model summary across all 17 datasets. IRT rank shown; "–" = model absent. Accuracy and IRT ability ($\hat{\theta}$) shown where available.

Dataset	M	GNN	GNN+T2.5	GatedGNN	RelGT	Griffin
GNN ranks #1						
rel-amazon	35	**1** (.864)	–	–	–	–
rel-arxiv	35	**1** (.713)	–	–	–	–
rel-trial	35	**1** (.624)	–	–	–	–
dbinfer-stackex.	36	**1** (.972)	36 (.045)	–	–	–
GNN mid-table						
rel-ratebeer	34	3 (.875)	–	–	–	–
rel-f1	39	7 (.621)	38 (.522)	39 (.502)	37 (.541)	10 (.612)
dbinfer-outbrain	36	7 (.807)	36 (.724)	–	–	–
dbinfer-amazon	34	5 (.664)	2 (.664)	3 (.664)	–	–
dbinfer-seznam	36	12 (.939)	10 (.939)	–	–	–
dbinfer-diginetica	37	28 (.990)	36 (.000)	37 (.000)	–	–
GNN absent / ranks low						
rel-avito	35	35 (.854)	–	–	–	31 (.881)
rel-event	36	–	34 (.427)	36 (.425)	–	–
rel-hm	35	–	–	–	–	35 (.413)
rel-stack	34	–	–	–	–	32 (.838)
rel-salt	33	–	–	–	–	–
dbinfer-avs	27	–	–	–	–	–
dbinfer-retail.	31	–	–	–	–	–

H. Difficulty Stratification (Extended)

Table 34. Accuracy by IRT difficulty tercile on **rel-f1** (selected models). Items split into Easy/Medium/Hard by 2PL difficulty β_i.

Model	Easy	Med	Hard	Gap	Family
LightGBM	.999	.763	.008	.990	GBT
TabPFN	.999	.808	.008	.990	Foundation
SVM	.999	.818	.059	.940	Linear
Dummy	.996	.778	.077	.919	Baseline
GNN	.997	.784	.082	.915	Graph
RDB+TabICL	.985	.753	.088	.897	RDBLearn
RDB+TabPFN	.959	.727	.133	.826	RDBLearn
XGBoost	.997	.623	.081	.916	GBT
Griffin	.874	.668	.292	.582	Graph
RelGT	.942	.401	.280	.663	Graph

I. Binarization Ablation

For regression tasks, we vary the binarization threshold (25th, 50th, 75th percentile of validation-set target) and refit IRT. Rankings are robust: Spearman $\rho > 0.85$ between IRT rankings at different thresholds on all tested datasets.

J. Structured Missingness Details

Table 35. Ranking correlation under structured missingness on rel-f1.

Missing %	ρ_{Accuracy}	ρ_{IRT}
10%	1.000	0.543
30%	0.629	0.596
50%	0.743	0.364
70%	0.093	0.261
90%	0.250	0.171

At low missingness (10–50%), accuracy preserves rankings better because simple averaging is a low-variance estimator. IRT's advantage emerges under severe missingness ($\geq$70%), where it leverages shared item-difficulty structure across items.

K. Weak-Signal Benchmark Analysis

Three datasets exhibit weak model-separation signal where accuracy provides minimal signal for model differentiation:

dbinfer-diginetica (37 models, accuracy spread < 0.001 among top 34 models): All models achieve $\sim$0.990 accuracy, with the top 34 separated by less than 0.001. IRT reveals a completely different ordering ($\rho = 0.011$): LightGBM jumps from accuracy rank #26 to IRT rank #1, while RDB+TabPFN drops from #3 to #34 ($\Delta = -31$). This suggests LightGBM handles the few difficult items better despite indistinguishable aggregate accuracy.

dbinfer-retailrocket (31 models, accuracy spread = 0.000): All 31 models achieve 0.974–0.975 accuracy. IRT $\rho = 0.210$ with bootstrap CI crossing zero $[-0.32, 0.36]$, confirming this benchmark cannot meaningfully distinguish models.

dbinfer-avs (27 models, accuracy spread = 0.014): All models cluster at 0.625–0.638 accuracy. IRT $\rho = 0.300$ with wide CI $[-0.18, 0.54]$. The ranking is unstable across bootstrap resamples.

Recommendation: These datasets should be flagged in benchmark suites. Item-level analysis can identify the subset of discriminating items for efficient evaluation.

L. Experimental Setup: Parameters, Libraries, and Settings

This section documents all software, parameters, and computational settings used to produce the results in this paper, to support full reproducibility.

L.1. Software and Libraries

All analyses were implemented in Python 3.8 on Azure ML. Table 36 lists the key libraries and their versions.

Table 36. Software dependencies and versions.

Library	Version / Notes
Python	3.8 (Azure ML environment)
NumPy	1.24.x
Pandas	2.0.x
SciPy	1.10.x (`scipy.stats.spearmanr`)
Matplotlib	3.7.x (`Agg` backend)
Seaborn	0.12.x (heatmaps)
`torch_measure`	Stanford CS321M package
PyTorch	2.1.x (IRT model fitting)
scikit-learn	1.3.x (baseline models)
XGBoost	2.0.x
LightGBM	4.1.x
CatBoost	1.2.x (GPU mode)
`relbench`	1.0.x (graph data loaders)
`rdblearn`	Deep Feature Synthesis

L.2. Data Source and Preprocessing

Data source. Model evaluation results were obtained from an exhaustive evaluation pipeline on the RelBench (Fey et al., 2023) and DBInfer benchmark suites, stored in `all_merged.pkl`. This file contains per-item binary responses for up to 39 models across 17 datasets and 55 tasks.

Binarization. All responses are binary: $Y_{ij} \in \{0, 1\}$, where 1 indicates the model's prediction matched the ground truth. For regression tasks, predictions were binarized at the per-task median.

Item cap. The number of items per task was capped at $n_I^{\max} = 50{,}000$. Tasks with fewer items used all available items.

Minimum model threshold. Tasks with fewer than `MIN_MODELS_PER_TASK` $= 10$ models were excluded

from analysis. For multi-task datasets, only models appearing in *all* tasks for that dataset were retained.

L.3. IRT Fitting Parameters

All IRT models were fit using `torch_measure` (Stanford CS321M, 2024). Table 37 summarizes the fitting configuration.

Table 37. IRT model fitting parameters.

Parameter	Value
Models fit	Rasch, 2PL, LogisticFM ($K = 1, 2, 4$)
Estimation method	MLE
Max epochs	200
Learning rate	0.05
Weight decay	0.01
Random seed	42
Train/test split	80/20 random item split
Bootstrap (Rasch & 2PL only)	
Bootstrap resamples	100
Bootstrap unit	Items (resampled with replacement)
CI method	2.5th / 97.5th percentile

For each dataset, a single response matrix $\mathbf{Y} \in \{0, 1\}^{M \times I}$ was constructed by stacking all tasks. Five IRT models were fit: **Rasch** (difficulty only), **2PL** (difficulty + discrimination), and **LogisticFM** with $K = 1, 2, 4$ latent factors. AUC was computed on 20% held-out items; Spearman ρ was computed between IRT ability rank and raw accuracy rank across all items.

L.4. Model Hyperparameters

Tables 38 to 40 list the complete hyperparameter settings for all 39 models evaluated. All models use a fixed random seed $S = 42$.

L.5. Computational Environment

Model training experiments ran on Azure ML compute clusters with NVIDIA H100 80GB GPUs (used by CatBoost, GNN, GatedGNN, RelGT, Griffin, and foundation models). Sklearn models (XGBoost, LightGBM, RandomForest, etc.) ran on CPU nodes within the same cluster. Total wall-clock time: $\sim$500 node-hours across 23 Azure ML jobs.

IRT fitting used `torch_measure` (Stanford CS321M, 2024) on a single GPU (NVIDIA A100 40GB), completing all 17 datasets (5 IRT models $\times$ 100 bootstrap resamples each) in under 20 minutes. Random seed: `SEED` $= 42$ for all stochastic operations.

L.6. Figure Generation

All figures use Matplotlib with `font.size=11`, `figure.dpi=200`, `font.family=serif`, and `savefig.bbox=tight`. Output format: PDF (vector) for paper figures, PNG (raster) for supplementary inspection. Bar charts use `steelblue` (Rasch) and `coral` (2PL). Heatmaps use `RdYlGn` diverging palette centered at zero.

L.7. Code and Reproducibility

Response matrices, IRT model results, and figure-generation code are available in the supplementary material. The complete pipeline, from raw evaluation outputs to all figures and tables, is contained in `run_irt_local.py` (491 lines).

Table 38. Sklearn model hyperparameters. Training set capped at 200,000 samples. Default parameters unless noted.

Model	Library	Key Hyperparameters
XGBoost	xgboost	n_estimators=100, max_depth=6, lr=0.1, eval_metric=logloss
XGB-deep	xgboost	n_estimators=200, max_depth=10, lr=0.1
XGB-shallow	xgboost	n_estimators=50, max_depth=3, lr=0.1
XGB-lr01	xgboost	n_estimators=200, max_depth=6, lr=0.01
XGB-lr3	xgboost	n_estimators=100, max_depth=6, lr=0.3
XGB-d4	xgboost	n_estimators=150, max_depth=4, lr=0.1
XGB-500	xgboost	n_estimators=500, max_depth=6, lr=0.05
LightGBM	lightgbm	n_estimators=100, max_depth=6, lr=0.1
LGBM-deep	lightgbm	n_estimators=200, max_depth=12, lr=0.1
LGBM-lr01	lightgbm	n_estimators=200, max_depth=6, lr=0.01
CatBoost	catboost	iterations=200, depth=6, lr=0.1, task_type=GPU
CB-deep	catboost	iterations=200, depth=10, lr=0.1, task_type=GPU
CB-lr01	catboost	iterations=300, depth=6, lr=0.01, task_type=GPU
RandomForest	sklearn	n_estimators=200, max_depth=10, n_jobs=-1
RF-deep	sklearn	n_estimators=200, max_depth=20, n_jobs=-1
RF-shallow	sklearn	n_estimators=200, max_depth=5, n_jobs=-1
ExtraTrees	sklearn	n_estimators=200, max_depth=10, n_jobs=-1
HistGradBoost	sklearn	max_iter=200, max_depth=6, lr=0.1
GradBoost	sklearn	n_estimators=100, max_depth=4, lr=0.1
AdaBoost	sklearn	n_estimators=100, lr=0.1
KNN	sklearn	n_neighbors=10, n_jobs=-1
LogReg	sklearn	max_iter=1000
ElasticNet	sklearn	penalty=elasticnet, solver=saga, l1_ratio=0.5
SVM	sklearn	kernel=rbf, probability=True
Dummy	sklearn	strategy=prior

Table 39. Foundation model and RDBLearn hyperparameters. Training set capped at 50,000 samples. Timeout: 900s (FMs), 7,200s (RDBLearn).

Model	Source	Key Settings
TabPFN	tabpfn	Pre-trained; flat-table features via simple joins; 50K sample cap
TabICL	tabicl	Pre-trained; flat-table features; 50K sample cap
ContextTab	contexttab	Pre-trained; flat-table features; 50K sample cap
TabSTAR	tabstar	Pre-trained; flat-table features; 50K sample cap
RealMLP	realmlp	Default hyperparameters; flat-table features
RDB+TabPFN	rdblearn	Deep Feature Synthesis (fastdfs) + TabPFN
RDB+TabICL	rdblearn	Deep Feature Synthesis + TabICL
RDB+CtxTab	rdblearn	Deep Feature Synthesis + ContextTab

Table 40. Graph model hyperparameters. All trained on the relational graph via relbench.modeling.

Model	Source	Key Settings
HeteroGraphSAGE	relbench	2-layer GraphSAGE, hidden=128, batch=512, epochs=10, lr=0.001
GatedGNN	relbench	2-layer GatedGraphConv, hidden=128, batch=512, epochs=10, lr=0.001
RelGT	relgt	Relational Graph Transformer, hidden=128, heads=4, layers=2
Griffin	griffin	Pre-trained graph foundation model; fine-tuned per task
GNN+TabPFN-2.5	relbench	GNN embeddings + TabPFN (v2.5) head
DBInfer	dbinfer	7 variants: {LightGBM, XGBoost, CatBoost, MLP} $\times$ {DFS configs}