ONE TABLE, MANY QUERIES: BENCHMARKING GENERALIST TABULAR MODELS ACROSS GENERATION, IMPUTATION, AND PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Tabular generation, missing-value imputation, and supervised prediction are usually developed and benchmarked as separate problems, even though each can be written as a conditional query to the same table distribution. This fragmentation leaves a basic question unresolved: does competence on one query carry to others, or does a multi-interface model merely expose several APIs? We introduce ONETABLEBENCH, to our knowledge the first benchmark designed to turn this shared probabilistic view into a unified empirical study. Rather than merging three leaderboards, ONETABLEBENCH preserves task-specific data, information, and evaluation contracts while aligning 56 datasets, 46 registered methods, and 20 semantic capability axes. Its evaluation forms a four-level evidence chain: auditing metric redundancy, reliability, and construct validity; measuring cross-task rank alignment on matched support; estimating transfer with matched interventions over all seven non-empty combinations of generation, imputation, and prediction objectives; and comparing fixed model identities with deployable specialist portfolios through Pareto and leave-one-dataset-out analyses. The resulting picture is neither universal transfer nor complete task independence. Metrics are compressible but remain multidimensional, and method rankings agree only along particular capability axes. Observational alignment does not produce a robust off-diagonal transfer effect under our prespecified gate; several joint-objective interactions are instead sub-additive, and classification and regression induce different objective frontiers. Deployment conclusions are similarly profile-dependent: evaluated generalists can surpass selected specialist portfolios on downstream utility, whereas the portfolios retain a clear advantage on categorical fidelity. ONETABLEBENCH therefore reframes tabular generality from binary task support or a single leaderboard rank into a capability profile over queries, contracts, and deployment priorities.

1 INTRODUCTION

Tabular learning is conventionally organized around three separate tasks. Generation asks whether synthetic rows preserve the distribution and utility of real data; imputation asks whether values removed under a missingness mechanism can be recovered; and prediction asks whether a designated target can be inferred from observed features. Each literature has developed its own datasets, baselines, metrics, and leaderboards (Yang et al., 2024; Khan et al., 2025; Erickson et al., 2025). This specialization has improved individual tasks, but it prevents a more fundamental assessment: whether the capabilities learned for one use of a table carry to other uses of the same table.

The three tasks are mathematically connected. Generation queries the full joint distribution, imputation queries arbitrary feature conditionals, and prediction queries the target conditional. Diffusion, autoregressive, arbitrary-conditioning, and foundation-model systems increasingly expose more than one of these interfaces (Ivanov et al., 2019; Jolicoeur-Martineau et al., 2024; Zhang et al., 2025b). Yet interface coverage is not evidence of generality: marginal fidelity can coexist with lost predictive dependencies, low imputation error with suppressed uncertainty, and accurate decisions with poor modeling of the feature distribution. Information contracts, finite capacity, approximate objectives, and task-specific inference can all break the expected connection.

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

We therefore change the unit of analysis from a task leaderboard to a *query-conditioned capability profile*. A query specifies what is observed, what is requested, and which data and evaluation contract applies. This perspective requires four levels of evidence: measurement validity, cross-query rank alignment, objective-level transfer or interference, and deployment against fixed specialist portfolios. To our knowledge, ONETABLEBENCH is the first benchmark to connect all four across generation, imputation, and prediction in one protocol-aware study.

The benchmark spans 56 datasets constructed from 492 candidates reported by 24 studies, 46 registered methods, and 20 retained semantic axes. Dataset selection is method-independent, the tasks share held-out splits, and information-access and missingness contracts remain explicit. We audit the metric space using common-support correlations, reliability analysis, and construct-specific stress tests before measuring rank alignment on matched method–dataset support. We then evaluate five model families on five datasets under all seven non-empty combinations of generation (G), imputation (I), and prediction (P) objectives, matched within backbone, dataset, and training seed. Fixed-identity Pareto and leave-one-dataset-out analyses finally compare generalists with specialist portfolios without post-hoc switching.

The resulting evidence rejects both simple extremes. The metric space is compressible but not one-dimensional: calibration, authenticity, dependency preservation, uncertainty, privacy risk, and downstream utility remain distinct. Rank agreement is capability-specific, with the broadest stable alignment between generation and imputation and substantially narrower, prediction-type-dependent relations elsewhere. Observational alignment does not become a robust off-diagonal transfer effect under the prespecified intervention gate; several joint-objective interactions are instead sub-additive, and classification and regression induce different objective frontiers. At deployment time, generalists can outperform selected specialist portfolios for some utility profiles, while specialization retains a clear advantage for categorical fidelity. Generality is therefore a profile over queries and contracts, not a binary interface label or a single rank.

Our contributions are:

- A new conditional-query formulation that makes data access, visibility, missingness, and evaluation contracts part of the capability being benchmarked;
- A broad, protocol-aware benchmark and measurement audit spanning three tasks and complementary fidelity, utility, robustness, uncertainty, calibration, privacy, and structural criteria;
- An evidence hierarchy that separates metric validity, observational rank alignment, matched-objective transfer, and fixed-identity deployment comparisons; and
- A multidimensional account of generality based on Pareto profiles and held-out specialist portfolios, rather than a universal score or an unattainable per-cell oracle.

2 PRELIMINARIES

2.1 DEFINITION

Table 1: The three tabular tasks expressed as probabilistic queries.

Task	Observed O	Queried Q	Target distribution
Generation	$\varnothing$	$\mathcal{V}$	$\mathbb{P}^*(Z)$
Imputation	$(\mathcal{I} \setminus M) \cup T_C$	M	$\mathbb{P}^*(Z_M \mid Z_O; C)$
Prediction	$\mathcal{I}$	$\{p+1\}$	$\mathbb{P}^*(Y \mid X; C)$

To compare methods within a common probabilistic framework, we formulate generation, imputation, and prediction as queries to the same joint distribution over a table. Let a complete row be $Z = (X_1, \ldots, X_p, Y) \in \mathcal{Z} := \left(\prod_{j=1}^{p} \mathcal{X}_j\right) \times \mathcal{Y}$, where $X = (X_1, \ldots, X_p)$ denotes the features, Y denotes the supervised target, and $\mathcal{Z}$ is the Cartesian product of their domains. Each variable may be continuous or discrete. We denote the unknown joint distribution of Z by $\mathbb{P}^*$. Let $\mathcal{I} = \{1, \ldots, p\}$ and $\mathcal{V} = \mathcal{I} \cup \{p+1\}$, with $Z_{p+1} = Y$. We represent each task as $q = (O, Q, C)$, where $O, Q \subseteq \mathcal{V}$, $O \cap Q = \varnothing$, and C specifies the task protocol. Given fitted parameters $\hat{\theta}$, the model approximates $\mathbb{P}^*(Z_Q \mid Z_O; C)$ by $\widehat{\mathbb{P}}_{\hat{\theta}}(Z_Q \mid Z_O; C)$. Table 1 summarizes the observed variables, queried variables,

and target distribution for each task. For imputation, $M \subseteq \mathcal{I}$ denotes the masked feature set, with $T_C = \{p+1\}$ when the target is visible under protocol C and $T_C = \varnothing$ otherwise.

2.2 Dataset Construction

Before evaluating any method, we construct a 56-dataset benchmark from 492 OpenML datasets (Vanschoren et al., 2013) reported across 24 studies. We apply prespecified feasibility filters, then use method-independent descriptors for PCA and k-means (Jolliffe, 2002; Lloyd, 1982). The nearest nonduplicate candidate to each of 50 centroids forms the OpenML track, while six TabStruct structural-causal-model datasets (Jiang et al., 2026) form a separate structural track. Dataset selection does not use results from any evaluated method. All tasks share a fixed train–test split, and model fitting, preprocessing, schema estimation, and model selection use the training split only. Appendix A details sources, matching, filtering, descriptors, clustering, duplicate suppression, aggregate benchmark composition, the fixed split, and the separate SCM track.

3 Evaluation Metrics

3.1 Common-support rank protocol

Let $x_{s,m,k}$ be the replicate-averaged value of metric k for method m in stratum s. We orient it using $\phi_k(x) \in \{x, -x, -|x - a_k|\}$ for higher-is-better, error, or target-proximity metrics. For collection c with members $\mathcal{K}_c$, let $\mathcal{M}_{s,c}$ contain methods with observations for every member:

$$r_{s,m,k} = \mathrm{prank}_{m' \in \mathcal{M}_{s,c}}(\phi_k(x_{s,m,k})), \qquad S_{s,m,c} = \frac{1}{|\mathcal{K}_c|} \sum_{k \in \mathcal{K}_c} r_{s,m,k}. \tag{1}$$

Ties receive the average percentile rank, and $S_{s,m,c}$ requires all declared members. A metric pair $c = \{i, j\}$ is used to audit redundancy; a retained axis c defines an axis score. Both use the same orientation and common support within separate protocol strata; full details are in Appendix C. All subsequent analyses keep the same fixed member set $\mathcal{K}_c$ and the same member-complete, equal-weight aggregation rule. Their ranking reference populations remain analysis-specific: for example, Task Rank Analysis ranks within a task pair's aligned common-method support, whereas Controlled Transfer ranks within the seven objective arms.

3.2 Redundancy and retained axes

Table 2: Metric-space redundancy by task.

Cohort	Metrics	Clusters	PCs	PCA mass (PC1)
Generation	29	17	5	80% (53%)
Imputation	10	8	3	74% (41%)
Prediction (cls.)	10	4	2	88% (75%)
Prediction (reg.)	5	2	1	86% (86%)

Redundancy occurs in every task but varies in extent: generation retains the richest local structure, whereas prediction shows the greatest compression (Table 2). PCA shows stronger global compression, but a shared latent direction does not imply that two metrics are substitutable.

The audit therefore informs axis construction without mechanically determining it. We merge only aligned metrics that share an estimand, preserve distinct or opposing quantities, and keep the components of diagnostic bundles available for separate reporting. This yields 20 semantic axes; protocol strata remain separate rather than being counted as new constructs. Table 3 records their exact membership; Appendix C gives the task-specific rationales alongside Figures 9– 10. Axis scores remain separate rather than forming a universal score.

Where applicable, our metric definitions follow established constructions. Marginal-shape errors combine per-column two-sample KS statistics for numerical variables (Smirnov, 1948) with categorical total-variation distances. We also use C2ST (Lopez-Paz & Oquab, 2017); α-precision,

β-recall, and authenticity (Alaa et al., 2022); a mixed-type k-way query workload adapted from discrete marginal queries (McKenna et al., 2019); and a TSTR gap adapted from the TSTR protocol (Esteban et al., 2017). The privacy axis combines benchmark-defined closest-record diagnostics based on Gower distance (Gower, 1971), a distance-based membership diagnostic motivated by membership-inference attacks and low-FPR reporting (Shokri et al., 2017; Carlini et al., 2022), and a legacy δ-presence-inspired diagnostic (Nergiz et al., 2007); none constitutes a privacy guarantee. The imputation distribution axes use Jensen–Shannon divergence (Lin, 1991) and the 1-Wasserstein distance (Villani, 2009). Classification evaluation includes balanced accuracy and MCC (Brodersen et al., 2010; Matthews, 1975), ROC-AUC and average precision (Hanley & McNeil, 1982; Davis & Goadrich, 2006), proper scores (Brier, 1950; Good, 1952), and calibration and selective-prediction measures (Naeini et al., 2015; Guo et al., 2017; Geifman et al., 2019).

Table 3: **Retained capability axes and exact metric membership.** Within an axis, direction-aligned metrics are equally averaged after within-protocol ranking (Equation 1); diagnostic bundle components remain reportable.

Task	Retained axes (member metrics)
G	**Fidelity**: `numerical_ks_error`, `shape_error`, `one_way_error`, `two_way_error`, `three_way_error`; **C2ST**: `gbdt_c2st_auroc`, `logistic_c2st_auroc`; **α-precision**: `alpha_precision`; **association**: `mixed_association_error`, `mixed_association_similarity`; **TSTR gap**: `tstr_gap`; **global-utility compromise**: `global_utility_ratio`, `beta_recall_coverage`, `authenticity`; **privacy-risk diagnostics**: `mean_min_euclidean_dcr`, `normalized_median_gower_dcr`, `distance_mia_auroc`, `distance_mia_tpr_at_1pct_fpr`, `delta_presence`.
I	**numerical point accuracy**: `macro_nmae`, `macro_nrmse`; **categorical point accuracy**: `categorical_balanced_accuracy`, `categorical_macro_f1`; **categorical distribution**: `categorical_jsd`; **numerical distribution**: `numeric_wasserstein1`; **association**: `association_matrix_error`; **downstream utility gap**: `utility_gap`.
P (cls.)	**hard-decision performance**: `accuracy`, `balanced_accuracy`, `macro_f1`, `mcc`; **proper scores**: `brier_score`, `log_loss`; **ranking/selective risk**: `roc_auc`, `average_precision`, `aurc`; **calibration**: `ece`.
P (reg.)	**squared-error family**: `r2`, `nrmse`, `rmse`; **mean absolute error**: `mae`; **median error**: `median_absolute_error`.

4 Task Rank Analysis

4.1 Objective and Method

To characterize cross-task alignment among existing methods, we compare their relative rankings across generation (G), imputation (I), and prediction (P). Consistent rankings would indicate that the evaluated capabilities covary across tasks, whereas axis-dependent or conflicting rankings would show that cross-task performance cannot be adequately summarized by a single score.

For each task pair a and b, we identify the common method pool $\mathcal{M}_{ab}$, align metric directions, average scores over seeds or replicates, and analyze classification and regression separately. Imputation uses the target-observed, missing-completely-at-random (MCAR) setting (Rubin, 1976) with a missing rate of 0.3.

Capability-axis scores follow the common-support rank protocol in Section 3.1. For each axis pair c, c' and dataset d, we retain methods in $\mathcal{M}_{ab}$ for which all declared members of both axes are valid. We orient the metrics and compute percentile ranks on this aligned method set, then aggregate the member ranks according to Equation 1. The ranking reference set is specific to each task pair and axis pair.

We compute one pooled Spearman correlation (Spearman, 1904) over the aligned dataset–method units. For every single-method deletion, support, ranks, and axis scores are recomputed from scratch. An axis pair is *LOMO-direction-preserving* only when every effective deletion preserves the sign of the full-pool correlation; deletions that change no valid unit are excluded from this diagnostic. Appendix D gives the full construction.

4.2 Results and Discussion

Cross-task rank agreement depends on the capability axes being compared (Figure 1). No uniform correspondence holds across all tasks and evaluation dimensions.

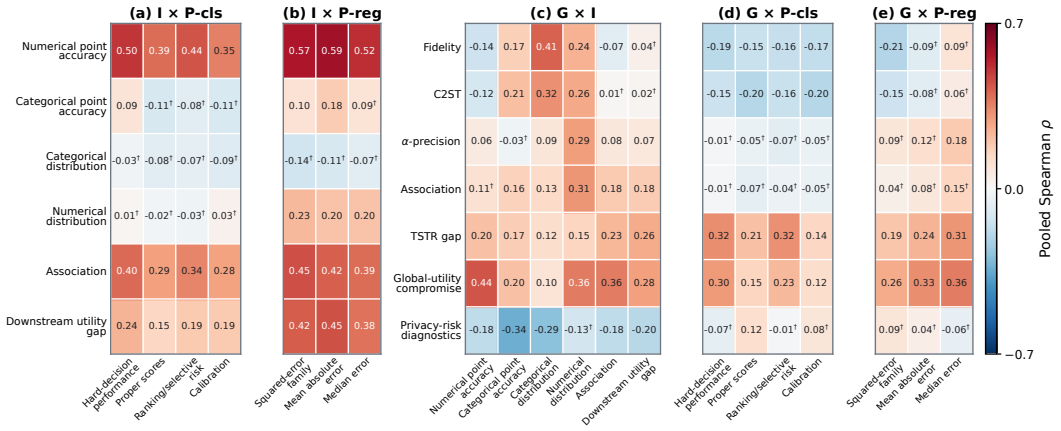


Figure 1: **Cross-task rank consistency across retained capability axes.** Each cell reports the pooled Spearman correlation between member-complete capability-axis scores computed over the same aligned dataset–method support. Red and blue indicate positive and negative correlations. † marks axis pairs for which at least one effective single-method deletion changes the sign of the full-pool correlation or produces zero correlation. G, I, P-cls, and P-reg denote generation, imputation, classification prediction, and regression prediction.

Imputation–prediction. Between imputation and classification prediction, 13 of the 24 axis pairs are LOMO-direction-preserving. Numerical point accuracy, association, and downstream utility gap show positive, direction-preserving correlations with all four classification axes; categorical point accuracy does so only with hard-decision performance. Between imputation and regression prediction, 14 of 18 pairs preserve direction. Besides all pairs involving numerical point accuracy, numerical distribution, association, or downstream utility gap, categorical point accuracy has positive, direction-preserving correlations with the squared-error and mean-absolute-error axes.

Generation–imputation. This task pair has the largest share of LOMO-direction-preserving relationships, 36 of 42. TSTR gap and global-utility compromise have positive, direction-preserving correlations with all six imputation axes. Privacy-risk diagnostics have negative, direction-preserving correlations with five axes, the exception being numerical distribution. The remaining generation axes retain mixed signs, so even this relatively robust task pair has no single ordering.

Generation–prediction. For classification prediction, 17 of the 28 axis pairs are LOMO-direction-preserving. Fidelity and C2ST correlate negatively with all four classification axes, whereas TSTR gap and global-utility compromise correlate positively with all four; privacy-risk diagnostics have one positive, direction-preserving correlation with proper scores. For regression prediction, 9 of 21 pairs preserve direction: all six TSTR-gap and global-utility pairs, the squared-error pairs for Fidelity and C2ST, and α-precision with median error.

Overall. Unified modeling does not produce a task-invariant method ranking in our benchmark. The strength and direction of these correlations depend on the capability axes. We therefore use axes as the unit of comparison and do not construct an overall cross-task score.

5 From Rank Alignment to Causal Transfer

Rank alignment is observational: two tasks may rank methods similarly because they favor the same architectures or reuse related evaluation criteria, even when optimizing one task does not improve the other. We therefore complement Section 4 with a controlled intervention on the training objective. This addresses two questions that a correlation analysis cannot answer: whether adding one task objective improves a different recipient task, and whether objectives combine additively within a shared model.

5.1 MATCHED OBJECTIVE INTERVENTIONS

We train Gaussian Copula (Patki et al., 2016), Forest-VP Joint (Jolicoeur-Martineau et al., 2024), TabDiff (Shi et al., 2025), TabSyn (Zhang et al., 2024), and TabDDPM (Kotelnikov et al., 2023) under the seven non-empty objective arms $\mathcal{C} = \{G, I, P, GI, GP, IP, GIP\}$. The five evaluation datasets are listed in Appendix E. Comparisons are matched within backbone, dataset, training seed, and atomic evaluation setting, so the objective arm is the only intervention.

The analysis follows the measurement contract in Section 3. Each metric member is oriented and ranked across the seven arms within a block, with ties receiving the average rank. The member ranks are then averaged within their semantic axis. An axis is formed only when every declared member is finite for every arm. Imputation mechanism–rate settings remain atomic through matching and receive equal weight only after their contrasts have been formed. For source objective s, recipient t, remaining task u, and axis a, the recipient-conditioned effect is

$$\Delta_{s\to t,a} = \frac{1}{2}\left[(S_{st,a} - S_{t,a}) + (S_{stu,a} - S_{tu,a})\right], \qquad \{s,t,u\} = \{G,I,P\}. \tag{2}$$

The recipient objective is present on both sides, while the third objective is averaged over absent and present contexts. Intervals use a crossed backbone–dataset bootstrap with seed resampling. We reserve a robust marker for effects that jointly pass uncertainty, multiplicity, practical-size, and cross-backbone direction checks; Appendix E gives the complete gate.

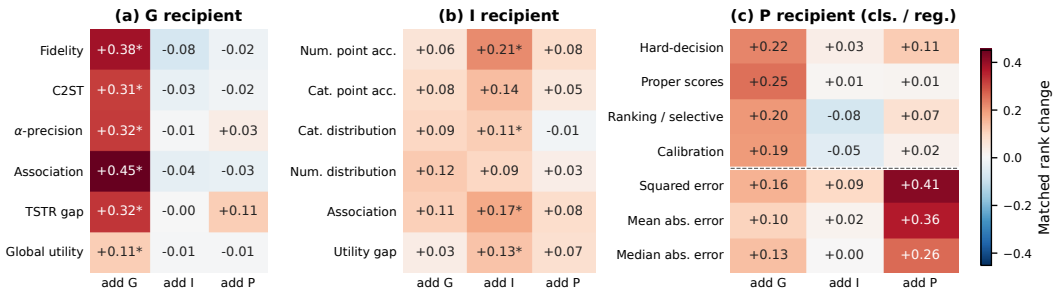


Figure 2: **Matched objective effects on the retained capability axes.** Panels (a)–(c) report changes on the normalized seven-arm rank scale. Diagonal cells are self-inclusion manipulation checks; off-diagonal cells are cross-task effects. Classification and regression prediction remain separate, and * marks passage of the prespecified robustness gate.

5.2 CROSS-TASK GAINS DO NOT SURVIVE THE ROBUSTNESS GATE

Figure 2 shows that adding the generation and imputation objectives improves multiple intended recipient axes. In contrast, under the primary within-backbone seven-arm contract, no retained-axis off-diagonal source–recipient effect passes the complete robustness gate. Thus the observational alignment in Section 4 does not establish a transfer edge that is stable across the evaluated backbones and datasets once architecture, data, and seed are held fixed. This is an absence of robust evidence under the declared criterion, not evidence that every backbone-level effect is zero.

Figure 3 uses the P→G utility decomposition to illustrate the distinction. The retained TSTR axis has a positive point estimate of 0.111 $[-0.012, 0.249]$, whereas the global-utility composite is -0.007 $[-0.047, 0.042]$. A scope-matched per-column evaluator gives the same qualitative pattern: the target-column estimate is 0.078, while the all-column and non-target estimates are near zero. Yet the direct target-minus-all and target-minus-non-target contrasts are 0.086 $[-0.039, 0.244]$ and 0.087 $[-0.057, 0.265]$, respectively. The point estimates are compatible with target-aligned transfer, but neither the transfer nor its localization is stable across backbones.

5.3 OBJECTIVE INTERACTIONS AND PARETO FRONTIERS

Cross-task main effects need not reveal interference between objectives. We therefore estimate anchor-conditional differences-in-differences; for example,

$$\Delta_{G\times I|P,a} = (S_{GIP,a} - S_{GP,a}) - (S_{IP,a} - S_{P,a}). \tag{3}$$

Zero denotes additivity, while a negative value means that the incremental benefit of adding I is smaller when G is already present. This contrast defines a P-anchored interaction within the seven-arm design.

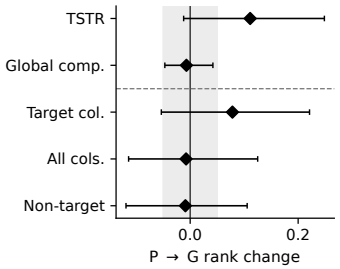


Figure 3: **P→G utility by scope.** Pooled effects with 95% bootstrap intervals; gray marks ± 0.05 rank units. Localization uses the paired contrasts in the text.

Table 4: **Core evidence from the objective intervention.** Effects are normalized seven-arm rank changes. Panel (a) lists every robust P-anchored G–I interaction with crossed-bootstrap 95% CIs; panel (b) gives the jointly computed three-dimensional objective-arm Pareto sets.

(a) Robust conditional interactions

Interaction	Recipient axis	Effect [95% CI]	p_{Holm}
$G \times I \mid P$	G C2ST	$-.320$ $[-.527, -.117]$	.0096
$G \times I \mid P$	G fidelity	$-.356$ $[-.589, -.103]$	.0096
$G \times I \mid P$	G global utility	$-.119$ $[-.224, -.035]$	.0275
$G \times I \mid P$	I association	$-.449$ $[-.636, -.223]$	.0001
$G \times I \mid P$	I numerical point accuracy	$-.526$ $[-.703, -.230]$	.0108

(b) Objective-arm Pareto sets

Prediction track	Exact Pareto set	$\epsilon = .05$ set
Classification	$\{G, GP\}$	$\{G, GP, GIP\}$
Regression	$\{G, GP, IP\}$	$\{G, GP, IP\}$

The clearest effect of the controlled intervention is negative interaction rather than positive cross-task transfer. $G \times I \mid P$ is robustly sub-additive for generation C2ST, fidelity, and global utility, and for imputation association and numerical point accuracy (Table 4). These effects span distributional, relational, and point-recovery criteria, so they are not a single-metric artifact. They also do not imply that G or I is independently harmful: they show that their combined increment is smaller in the implemented P-conditioned mixture.

We then compare the seven objective arms in a three-task Pareto analysis. Within each matched block, semantic axes receive equal weight in G, I, and P task macro-scores; classification and regression define separate geometries. Exact dominance allows no loss on any coordinate and requires a strict gain on one; material dominance instead requires a gain of at least $\epsilon = 0.05$ with no loss on the others. Under strict member-complete support, the classification exact frontier is $\{G, GP\}$ and its material frontier is $\{G, GP, GIP\}$. The regression exact and material frontiers are both $\{G, GP, IP\}$. No GIP-versus-singleton generality gap passes the robustness gate. Hence G and GP belong to both the classification and regression frontiers, while these separately estimated frontiers retain different additional arms; training on all three objectives is not a universally preferred policy. Full projections and stability analyses appear in Appendix E. This objective-level result motivates the next section's comparison of fixed, deployable method identities.

6 Fixed-Method Generality

The preceding sections separate observational alignment from objective transfer. We now ask whether one registered generalist can replace fixed single-task methods across the retained capabilities. Every candidate keeps one canonical method identity across datasets, axes, and missingness settings; neither side may assemble an oracle from different identities. Comparisons use the exact members of Table 3 and the within-protocol percentile composites in Equation 1. Exact and $\epsilon = 0.05$ Pareto relations are computed jointly. For cross-task displays only, imputation is summarized by the direct equal-weight mean of its twelve separately ranked target-observed settings. These profile and portfolio views are therefore deployment diagnostics; the strict estimands and setting-specific results are in Appendix F. The version-specific foundation-model identities are TabPFN-3, TabICLv2, and LimiX-2 (Grinsztajn et al., 2026; Qu et al., 2026; Zhang et al., 2026).

Generality is capability-specific. For generalist g, dataset d, and axis a, let B_{gda} be the fraction of contract-compatible fixed specialists beaten by g, with half credit for a tie. Figure 4 averages B_{gda} over paired datasets; for imputation the comparison is performed within each setting before the twelve setting summaries are averaged. The result is a task-relative position, not an overall score.

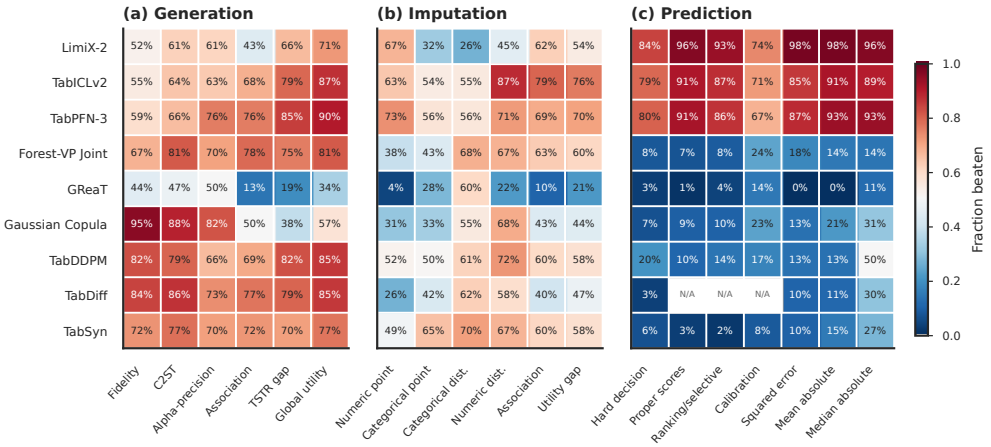


Figure 4: **A generalist's advantage is capability-specific.** Cells are the mean fraction of fixed specialists beaten on one axis. Panel (a) is generation, panel (b) equally averages the twelve separately evaluated target-observed imputation settings, and panel (c) is prediction. Columns are never averaged into a leaderboard.

The ordering changes sharply across axes. Gaussian Copula is strongest on several generation-distribution criteria, whereas TabPFN-3 leads generation utility; imputation leaders differ across the numerical, categorical, association, and utility axes. LimiX-2 leads every displayed prediction axis, but not the corresponding generation and imputation comparisons. No fixed identity is best everywhere: supporting three interfaces and being uniformly competitive are empirically different properties.

Capability profiles and Pareto frontiers. To make joint trade-offs interpretable, we use three prespecified G–I–P profiles rather than average all axes. Numerical fidelity combines generation fidelity, numerical imputation point accuracy, and regression squared error; categorical fidelity combines generation fidelity, categorical imputation point accuracy, and classification hard-decision performance; downstream utility combines generation TSTR, imputation utility preservation, and classification hard-decision performance. The top row of Figure 5 shows all three coordinates of each fixed generalist identity. Only LimiX-2 and TabDDPM remain on the practical numerical-fidelity frontier. The practical categorical-fidelity frontier contains Gaussian Copula, TabDDPM, TabDiff, and TabPFN-3, while the utility frontier retains LimiX-2, TabDDPM, TabICLv2, and TabPFN-3. The identity-aware Pareto set therefore depends on the capability profile. Frontier membership alone does not show the trade-offs visible in the three coordinates.

Held-out comparison with specialist portfolios. A deployable specialist alternative is one fixed triple (m_G, m_I, m_P), not a method selected after observing a test case. In each leave-one-dataset-out (LODO) fold, we choose a maximin triple on the remaining datasets, freeze it, and compare its held-out worst-task percentile with each generalist. The bottom row of Figure 5 gives the paired gaps. For numerical fidelity, the selected portfolio has a clear advantage over Forest-VP Joint, Gaussian Copula, TabDDPM, TabDiff, and TabSyn; the remaining comparisons are unresolved. For categorical fidelity, every generalist has a lower mean held-out worst-task percentile than the selected portfolio, with gaps ranging from -0.335 to -0.200 and all displayed bootstrap intervals below zero. For utility, TabICLv2 exceeds the portfolio by 0.178 (95% CI $[0.110, 0.244]$) and TabPFN-3 by 0.164 ($[0.078, 0.245]$), whereas LimiX-2 and TabDDPM remain unresolved and the portfolio is better for the other four methods.

LODO compares held-out datasets using a benchmark-wide candidate set; folds overlap, and the bootstrap intervals condition on the selected portfolios. Together, the two figures characterize generality as a capability profile under a declared deployment preference, rather than a binary interface label or a universal ranking.

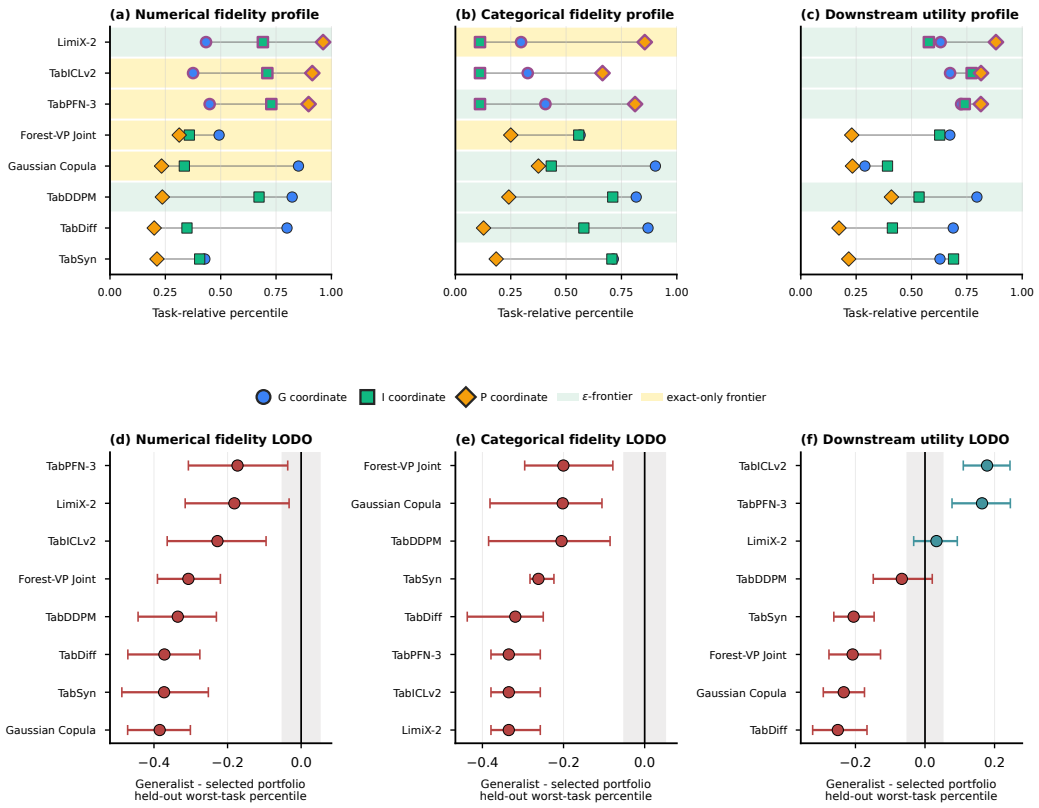


Figure 5: **Fixed identities show profile-dependent deployment trade-offs.** Panels (a)–(c) show numerical fidelity, categorical fidelity, and downstream utility, respectively. They connect each generalist's G, I, and P percentiles and share the method order shown at left; green and yellow rows mark the practical and exact-only three-dimensional frontiers. Panels (d)–(f) report the corresponding LODO generalist-minus-selected-portfolio difference in held-out worst-task percentile, with whole-dataset bootstrap intervals and a gray ± 0.05 band. Positive values favor the generalist.

7 Conclusion

OneTableBench recasts generation, imputation, and prediction as a family of conditional queries to one table distribution while preserving the contracts that make each query meaningful. It links measurement auditing, observational alignment, matched-objective intervention, and fixed-identity deployment in one evidence chain spanning 56 datasets, 46 registered methods, and 20 semantic capability axes. This broader view produces a qualified but consistent picture. Metrics can be compressed but remain multidimensional; rankings align only along particular axes and do not establish causal transfer. No off-diagonal effect passes the primary robustness gate, several interactions are sub-additive, and classification and regression yield different frontiers. Deployment is also profile-dependent: utility can favor a generalist, whereas categorical fidelity favors specialization. The tasks are therefore neither identical nor unrelated: generality depends on the query, contract, capability, and deployment preference. OneTableBench provides a new empirical language for future unified models—demonstrated capability rather than nominal support, controlled transfer rather than correlation for causal claims, and explicit profiles rather than a universal rank. Appendix G details the remaining scope.

AI USE STATEMENT

We used AI tools to assist with manuscript drafting and editing, figure preparation, and LaTeX formatting. The authors are responsible for the final manuscript and its scientific claims.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we describe the task definitions in Section 2 and the evaluation and aggregation protocols in Section 3. Dataset sources, selection criteria, data splits, and preprocessing procedures are detailed in Appendix A. Baseline configurations, hyperparameter settings and selection rules, random seeds, and computational resources are provided in Appendix H. Metric definitions and result aggregation procedures are specified in Appendices B and C, with analysis-specific protocols in Appendices D–F. Our evaluation code, configuration files, and scripts for reproducing the reported results are included in the supplementary material.

ETHICS STATEMENT

This study evaluates tabular learning methods using publicly available datasets under their respective licenses and terms of use. We do not collect additional personal data or attempt to identify individuals in the datasets. We recognize that synthetic data may retain sensitive information from the training data, and that empirical privacy evaluations do not constitute formal privacy guarantees. Furthermore, aggregate benchmark performance does not establish fairness across demographic groups or suitability for deployment in high-stakes settings. Any downstream use should therefore be accompanied by application-specific assessments of privacy, fairness, and reliability.

REFERENCES

Ahmed Alaa, Boris van Breugel, Evgeny Saveliev, and Mihaela van der Schaar. How faithful is your synthetic data? sample-level metrics for evaluating and auditing generative models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 290–320, 2022. URL https://proceedings.mlr.press/v162/alaa22a.html.

Edesio D. Alcobaça, Felipe Siqueira, Adriano Rivolli, Luís P. F. Garcia, Jefferson T. Oliva, and André C. P. L. F. de Carvalho. MFE: Towards reproducible meta-feature extraction. *Journal of Machine Learning Research*, 21(111):1–5, 2020. URL https://jmlr.org/papers/v21/19-348.html.

Vadim Borisov, Kathrin Seßler, Tobias Leemann, Martin Pawelczyk, and Gjergji Kasneci. Language models are realistic tabular data generators. In *International Conference on Learning Representations*, 2023.

João Bravo. NRGBoost: Energy-based generative boosted trees. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=wQHyjIZ1SH.

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

Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. *Classification and Regression Trees*. Wadsworth, 1984.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. doi: 10.1175/1520-0493(1950)078⟨0001:VOFEIT⟩2.0.CO;2.

Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M. Buhmann. The balanced accuracy and its posterior distribution. In *International Conference on Pattern Recognition*, pp. 3121–3124, 2010. doi: 10.1109/ICPR.2010.764.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. Membership inference attacks from first principles. In *IEEE Symposium on Security and Privacy*, pp. 1897–1914, 2022. doi: 10.1109/SP46214.2022.9833649.

Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002. doi: 10.1613/jair.953. URL https://www.cs.cmu.edu/afs/cs/project/jair/pub/volume16/chawla02a.pdf.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

Thomas M. Cover and Peter E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967. doi: 10.1109/TIT.1967.1053964.

Harald Cramér. *Mathematical Methods of Statistics*. Princeton University Press, 1946.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and ROC curves. In *International Conference on Machine Learning*, pp. 233–240, 2006. doi: 10.1145/1143844.1143874.

Tianyu Du, Luca Melis, and Ting Wang. ReMasker: Imputing tabular data with masked autoencoding. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2309.13793.

Yuntao Du and Ninghui Li. Systematic assessment of tabular data synthesis. In *ACM SIGSAC Conference on Computer and Communications Security*, pp. 2414–2428, 2025. doi: 10.1145/3719027.3765067. URL https://arxiv.org/abs/2402.06806.

Tony Duan, Avati Anand, Daisy Yi Ding, Khanh K. Thai, Sanjay Basu, Andrew Ng, and Alejandro Schuler. NGBoost: Natural gradient boosting for probabilistic prediction. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 2690–2700, 2020. URL https://proceedings.mlr.press/v119/duan20a.html.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://papers.neurips.cc/paper_files/paper/2019/hash/7ac71d433f282034e088473244df8c02-Abstract.html.

Bradley Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1): 1–26, 1979. doi: 10.1214/aos/1176344552.

Nick Erickson, Lennart Purucker, Andrej Tschalzev, David Holzmüller, Prateek Desai, David Salinas, and Frank Hutter. TabArena: A living benchmark for machine learning on tabular data. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/1697e3fb412da11dc9488249f9e7bbc9-Abstract-Datasets_and_Benchmarks_Track.html.

Cristóbal Esteban, Stephanie L. Hyland, and Gunnar Rätsch. Real-valued (medical) time series generation with recurrent conditional GANs. *arXiv preprint arXiv:1706.02633*, 2017. URL https://arxiv.org/abs/1706.02633.

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. URL https://arxiv.org/abs/2510.02625.

Yonatan Geifman, Guy Uziel, and Ran El-Yaniv. Bias-reduced uncertainty estimation for deep neural classifiers. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SJfb5jCqKm.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. URL https://doi.org/10.1198/016214506000001437.

Irving John Good. Rational decisions. *Journal of the Royal Statistical Society: Series B*, 14(1): 107–114, 1952.

Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://arxiv.org/abs/2106.11959.

Yury Gorishniy, Akim Kotelnikov, and Artem Babenko. TabM: Advancing tabular deep learning with parameter-efficient ensembling. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2410.24210.

John C. Gower. A general coefficient of similarity and some of its properties. *Biometrics*, 27(4): 857–871, 1971. doi: 10.2307/2528823.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/0378c7692da36807bdec87ab043cdadc-Abstract-Datasets_and_Benchmarks.html.

Léo Grinsztajn et al. TabPFN-3: Technical report. *arXiv preprint arXiv:2605.13986*, 2026. URL https://arxiv.org/abs/2605.13986.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330, 2017. URL https://proceedings.mlr.press/v70/guo17a.html.

Xizewen Han, Huangjie Zheng, and Mingyuan Zhou. CARD: Classification and regression diffusion models. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://arxiv.org/abs/2206.07275.

James A. Hanley and Barbara J. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29–36, 1982. doi: 10.1148/radiology.143.1.7063747.

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

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

Brian Hopkins and John G. Skellam. A new method for determining the type of distribution of plant individuals. *Annals of Botany*, 18(2):213–227, 1954. doi: 10.1093/oxfordjournals.aob.a083391.

Oleg Ivanov, Michael Figurnov, and Dmitry Vetrov. Variational autoencoder with arbitrary conditioning. In *International Conference on Learning Representations*, 2019.

Daniel Jarrett, Bogdan C. Cebere, Tennison Liu, Alicia Curth, and Mihaela van der Schaar. HyperImpute: Generalized iterative imputation with automatic model selection. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9916–9937, 2022. URL https://proceedings.mlr.press/v162/jarrett22a.html.

Xiangjian Jiang, Nikola Simidjievski, and Mateja Jamnik. TabStruct: Measuring structural fidelity of tabular data. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.11950.

Alexia Jolicoeur-Martineau, Kilian Fatras, and Tal Kachman. Generating and imputing tabular data via diffusion and flow-based gradient-boosted trees. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 1288–1296, 2024. URL https://proceedings.mlr.press/v238/jolicoeur-martineau24a.html.

Ian T. Jolliffe. *Principal Component Analysis*. Springer, 2 edition, 2002. doi: 10.1007/b98835.

Henry F. Kaiser. The application of electronic computers to factor analysis. *Educational and Psychological Measurement*, 20(1):141–151, 1960. doi: 10.1177/001316446002000116.

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

Falaah Arif Khan, Denys Herasymuk, Nazar Protsiv, and Julia Stoyanovich. Still more shades of null: An evaluation suite for responsible missing value imputation. *Proceedings of the VLDB Endowment*, 18(9):2899–2913, 2025. doi: 10.14778/3746405.3746416.

Akim Kotelnikov, Dmitry Baranchuk, Ivan Rubachev, and Artem Babenko. TabDDPM: Modelling tabular data with diffusion models. In *International Conference on Machine Learning*, 2023.

Anton Danholt Lautrup, Tobias Hyrup, Arthur Zimek, and Peter Schneider-Kamp. SynthEval: A framework for detailed utility and privacy evaluation of tabular synthetic data. *Data Mining and Knowledge Discovery*, 39(1), 2025. doi: 10.1007/s10618-024-01081-4. URL https://arxiv.org/abs/2404.15821.

Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991. doi: 10.1109/18.61115.

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

Stuart P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28 (2):129–137, 1982. doi: 10.1109/TIT.1982.1056489.

David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. In *International Conference on Learning Representations*, 2017. URL https://arxiv.org/abs/1610.06545.

Andrei Margeloiu, Xiangjian Jiang, Nikola Simidjievski, and Mateja Jamnik. TabEBM: A tabular data augmentation method with distinct class-specific energy-based models. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://arxiv.org/abs/2409.16118.

Brian W. Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta*, 405(2):442–451, 1975. doi: 10.1016/0005-2795(75)90109-9.

Duncan McElfresh, Sujay Khandagale, Jonathan Valverde, Vishak Prasad, Ganesh Ramakrishnan, Micah Goldblum, and Colin White. When do neural nets outperform boosted trees on tabular data? In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/f06d5ebd4ff40b40dd97e30cee632123-Abstract.html.

Leland McInnes, John Healy, and James Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018. URL https://arxiv.org/abs/1802.03426.

Ryan McKenna, Daniel Sheldon, and Gerome Miklau. Graphical-model based estimation and inference for differential privacy. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4435–4444, 2019. URL https://proceedings.mlr.press/v97/mckenna19a.html.

Mahdi Pakdaman Naeini, Gregory F. Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *AAAI Conference on Artificial Intelligence*, 2015.

Mehmet Ercan Nergiz, Maurizio Atzori, and Chris Clifton. Hiding the presence of individuals from shared databases. In *ACM SIGMOD International Conference on Management of Data*, pp. 665–676, 2007. doi: 10.1145/1247480.1247554.

Yidong Ouyang, Liyan Xie, Chongxuan Li, and Guang Cheng. MissDiff: Training diffusion models on tabular data with missing values. *arXiv preprint arXiv:2307.00467*, 2023. URL https://arxiv.org/abs/2307.00467.

Art B. Owen and Dean Eckles. Bootstrapping data arrays of arbitrary order. *The Annals of Applied Statistics*, 6(3):895–927, 2012. doi: 10.1214/12-AOAS547.

Neha Patki, Roy Wedge, and Kalyan Veeramachaneni. The synthetic data vault. In *IEEE International Conference on Data Science and Advanced Analytics*, pp. 399–410, 2016. doi: 10.1109/DSAA.2016.49.

Karl Pearson. *On the General Theory of Skew Correlation and Non-Linear Regression*. Dulau and Company, 1905. URL https://archive.org/details/galtonlab023.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. URL https://jmlr.org/papers/v12/pedregosa11a.html.

Bernhard Pfahringer, Hilan Bensusan, and Christophe G. Giraud-Carrier. Meta-learning by landmarking various learning algorithms. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pp. 743–750, 2000.

Prior Labs. tabpfn-extensions: Community extensions for TabPFN. GitHub repository, 2026. URL https://github.com/PriorLabs/tabpfn-extensions.

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

Zhaozhi Qian, Rob Davis, and Mihaela van der Schaar. Synthcity: A benchmark framework for diverse use cases of tabular synthetic data. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0140. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/09723c9f291f6056fd1885081859c186-Abstract.html.

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. URL https://arxiv.org/abs/2602.11139.

Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987. doi: 10.1016/0377-0427(87)90125-7.

Ivan Rubachev, Nikolay Kartashev, Yury Gorishniy, and Artem Babenko. TabReD: Analyzing pitfalls and filling the gaps in tabular deep learning benchmarks. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2406.19380.

Donald B. Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976. doi: 10.1093/biomet/63.3.581.

SDV. Sdmetrics documentation, 2026. URL https://docs.sdv.dev/sdmetrics/data-metrics/quality/kscomplement. Accessed 2026-09-26; exact benchmark adaptation documented in the metric dictionary.

Zineb Senane, Axel Karlsson, Lele Cao, Oleg Smirnov, Cheng Zhang, Sahar Asadi, Hedvig Kjellström, Gustav Eje Henter, and Ruibo Tu. Causality for tabular data synthesis: A high-order structure causal benchmark framework. *arXiv preprint arXiv:2406.08311*, 2024. URL https://arxiv.org/abs/2406.08311.

Juntong Shi, Minkai Xu, Harper Hua, Hengrui Zhang, Stefano Ermon, and Jure Leskovec. TabDiff: A mixed-type diffusion model for tabular data generation. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2410.20626.

Shohei Shimizu, Takanori Inazumi, Yasuhiro Sogawa, Aapo Hyvärinen, Yoshinobu Kawahara, Takashi Washio, Patrik O. Hoyer, and Kenneth Bollen. DirectLiNGAM: A direct method for learning a linear non-gaussian structural equation model. *Journal of Machine Learning Research*, 12:1225–1248, 2011. URL https://www.jmlr.org/papers/v12/shimizu11a.html.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy*, pp. 3–18, 2017. doi: 10.1109/SP.2017.41.

Patrick E. Shrout and Joseph L. Fleiss. Intraclass correlations: Uses in assessing rater reliability. *Psychological Bulletin*, 86(2):420–428, 1979. doi: 10.1037/0033-2909.86.2.420.

Thibault Simonetto, Salah Ghamizi, and Maxime Cordy. TabularBench: Benchmarking adversarial robustness for tabular deep learning in real-world use-cases. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-2492. URL https://arxiv.org/abs/2408.07579.

Nikolai Smirnov. Table for estimating the goodness of fit of empirical distributions. *The Annals of Mathematical Statistics*, 19(2):279–281, 1948. doi: 10.1214/aoms/1177730256.

Charles Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904. doi: 10.2307/1412159.

Daniel J. Stekhoven and Peter Bühlmann. Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2012.

Luís Torgo, Rita P. Ribeiro, Bernhard Pfahringer, and Paula Branco. SMOTE for regression. In *Progress in Artificial Intelligence*, pp. 378–389, 2013. doi: 10.1007/978-3-642-40669-0_33. URL https://web.cs.dal.ca/~ltorgo/publication/2013_trpb2013/.

Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B. Altman. Missing value estimation methods for DNA microarrays. *Bioinformatics*, 17(6):520–525, 2001. doi: 10.1093/bioinformatics/17.6.520.

Stef van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in R. *Journal of Statistical Software*, 45(3):1–67, 2011.

Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luís Torgo. OpenML: Networked science in machine learning. *ACM SIGKDD Explorations Newsletter*, 15(2):49–60, 2013. doi: 10.1145/2641190.2641198.

Mario Villaizán-Vallelado, Matteo Salvatori, Carlos Segura, and Ioannis Arapakis. Diffusion models for tabular data imputation and synthetic data generation. *ACM Transactions on Knowledge Discovery from Data*, 19(6), 2025. doi: 10.1145/3742435. URL https://arxiv.org/abs/2407.02549.

Cédric Villani. *Optimal Transport: Old and New*. Springer, 2009. doi: 10.1007/978-3-540-71050-9.

David S. Watson, Kristin Blesch, Jan Kapar, and Marvin N. Wright. Adversarial random forests for density estimation and generative modeling. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, 2023. URL https://arxiv.org/abs/2205.09435.

Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945. doi: 10.2307/3001968.

Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using conditional GAN. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://papers.nips.cc/paper_files/paper/2019/hash/254ed7d2de3b23ab10936522dd547b78-Abstract.html.

Scott Cheng-Hsin Yang, Baxter Eaves, Michael Schmidt, Ken Swanson, and Patrick Shafto. Structured evaluation of synthetic tabular data. *arXiv preprint arXiv:2403.10424*, 2024. URL https://arxiv.org/abs/2403.10424.

Han-Jia Ye, Si-Yang Liu, Hao-Run Cai, Qi-Le Zhou, and De-Chuan Zhan. A closer look at deep learning methods on tabular datasets. *arXiv preprint arXiv:2407.00956*, 2024. URL https://arxiv.org/abs/2407.00956.

Hengrui Zhang, Jiani Zhang, Balasubramaniam Srinivasan, Zhengyuan Shen, Xiao Qin, Christos Faloutsos, Huzefa Rangwala, and George Karypis. Mixed-type tabular data synthesis with score-based diffusion in latent space. In *International Conference on Learning Representations*, 2024.

Hengrui Zhang, Liancheng Fang, Qitian Wu, and Philip S. Yu. DiffPuter: Empowering diffusion models for missing data imputation. In *International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=3fl1SENSYO.

Xingxuan Zhang et al. LimiX: Unleashing structured-data modeling capability for generalist intelligence. *arXiv preprint arXiv:2509.03505*, 2025b. URL https://arxiv.org/abs/2509.03505.

Xingxuan Zhang et al. LimiX-2: A contextual mechanism network towards general structured-data intelligence. *arXiv preprint arXiv:2609.17488*, 2026. URL https://arxiv.org/abs/2609.17488.

Shuhan Zheng and Nontawat Charoenphakdee. Diffusion models for missing value imputation in tabular data. In *NeurIPS Workshop on Table Representation Learning*, 2022. URL https://arxiv.org/abs/2210.17128.

A Dataset Construction Details

A.1 Construction Objective and Overall Pipeline

The benchmark supports prediction, missing-value imputation, and data generation under a controlled computational budget. Evaluating every candidate dataset would be costly and would overrepresent sources with multiple versions and groups of structurally similar datasets. We therefore select representative datasets using only method-independent dataset properties and fixed landmarking statistics (Pfahringer et al., 2000). All selection rules and configurations are specified before method evaluation begins, so no evaluated method's performance influences dataset selection. Figure 6 summarizes the construction process, from the literature-derived candidate pool to the two final evaluation sets.

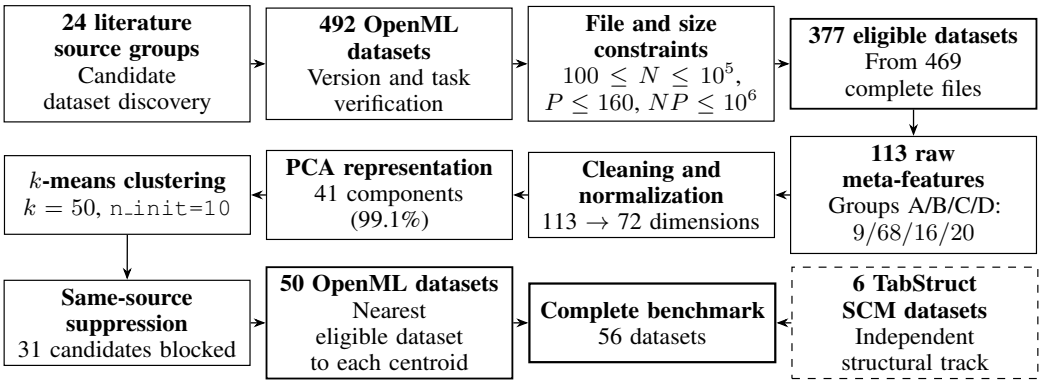


Figure 6: Dataset construction pipeline. The OpenML track uses prespecified size constraints, a meta-feature representation, and clustering to obtain 50 representative datasets. The 6 SCM datasets form a separate structural evaluation track.

A.2 Literature-Driven Candidate Dataset Pool

We identify candidate datasets from 24 deduplicated literature source groups. These sources cover general tabular prediction benchmarks as well as specialized studies of missing-value imputation, data generation, structural fidelity, and robustness. Table 5 lists these sources. ForestDiffusion covers both generation and imputation and is counted once; the two relevant TALENT studies form one source group. Literature sources define only the candidate pool and do not determine the weights used for representative selection.

After normalizing dataset names and common aliases, we resolve dataset identities using Dataset IDs reported in papers or official configurations, Task IDs that can be mapped to specific versions, frozen identifiers in official benchmark suites, and metadata that agree in dataset name, size, task type, and original source. We merge references to the same Dataset ID across studies while retaining all source attributions. Records with the same name but different Dataset IDs, versions, or sizes are not merged automatically. This process yields 492 datasets with unambiguous OpenML Dataset IDs (Vanschoren et al., 2013).

A.3 Data Parsing and Hard Filtering

Column types and target columns. Variable types follow the ARFF header rather than the appearance of the values. Nominal attributes are treated as categorical variables, whereas `numeric`, `real`, and `integer` attributes are treated as numerical variables. A `string` attribute is treated as categorical only when its cardinality does not exceed $\min(50, 0.2N_i)$. Other free-text fields and all `date`/`relational` attributes are removed. Thus, P_i counts usable features after removing target columns and unsupported fields, rather than raw CSV columns.

We use the OpenML `default_target_attribute` field to identify the target whenever available. If this field is missing, the target is inferred in order from common names, including `class`, `label`,

Table 5: The 24 deduplicated literature source groups used to construct the candidate dataset pool.

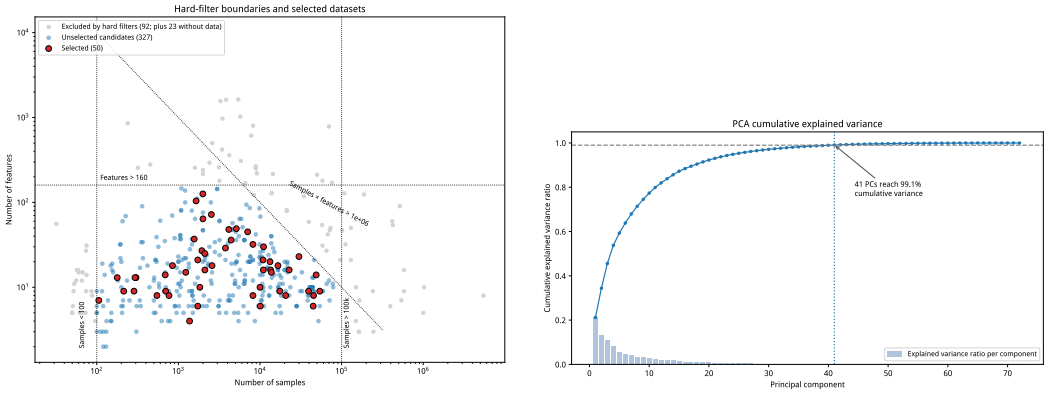
Category	Literature sources
Imputation and cross-task studies	Shades-of-Null (Khan et al., 2025); TabImpute (Feitelberg et al., 2025); ForestDiffusion (Jolicoeur-Martineau et al., 2024); DiffPuter (Zhang et al., 2025a); TabCSDI (Zheng & Charoenphakdee, 2022); MissDiff (Ouyang et al., 2023); ReMasker (Du et al., 2024); HyperImpute (Jarrett et al., 2022); TabGenDDPM (Villaizán-Vallelado et al., 2025).
Generation and structural evaluation	TabStruct (Jiang et al., 2026); Systematic Assessment of Tabular Data Synthesis (Du & Li, 2025); Synthcity (Qian et al., 2023); Structured Evaluation of Synthetic Tabular Data (Yang et al., 2024); CauTabBench (Senane et al., 2024); SynthEval (Lautrup et al., 2025); TabDDPM (Kotelnikov et al., 2023); TabDiff (Shi et al., 2025).
Prediction and robustness	TabArena (Erickson et al., 2025); TALENT (Liu et al., 2025; Ye et al., 2024); TabZilla (McElfresh et al., 2023); Why Trees (Grinsztajn et al., 2022); TabReD (Rubachev et al., 2025); CARD (Han et al., 2022); TabularBench (Simonetto et al., 2024).

`target`, `y`, and `outcome`, or from a low-cardinality final column. Such cases are marked by `target_inferred`. A numerical final column is inferred as a classification target only when it has between 2 and 30 unique values and the sample size is at least 20 times the number of unique values. For multi-target datasets, only the first target is used to compute prediction-related meta-features, while the original number of targets is retained.

Size constraints. To keep evaluation within a common computational budget, we impose the following constraints on each dataset $\mathcal{D}_i$:

$$100 \leq N_i \leq 100{,}000, \qquad P_i \leq 160, \qquad N_i P_i \leq 10^6. \tag{4}$$

The 492 source entries yield 469 datasets for size screening. The size constraints exclude 92 datasets, leaving 377 candidates. Figure 7(a) shows the filtering outcome in sample-size–feature-count space.



(a) Hard-filtering boundaries and selected datasets.

(b) PCA cumulative explained variance. Forty-one components reach the 99% threshold.

Figure 7: Dataset size constraints and PCA dimensionality reduction. In the left panel, gray points denote datasets removed by the size constraints, blue points denote eligible but unselected candidates, and red points denote the selected datasets.

A.4 Dataset Meta-Feature Representation

We recompute 113 meta-features for each candidate dataset using a unified local extractor. The extractor includes reproducible PyMFE statistics (Alcobaça et al., 2020) and standard clustering diagnostics (Hopkins & Skellam, 1954; Rousseeuw, 1987). The four meta-feature groups serve only

for interpretation and post hoc diagnostics; we assign no manual group weights. Table 6 reports the original counts and lists all dimensions retained after cleaning.

Table 6: Composition of the candidate-dataset meta-features and all 72 dimensions retained after cleaning.

Group	Original	Retained	Retained meta-features
A	9	9	`cat_feature_ratio`; `frac_binary_features`; `frac_numeric_lowcard`; `log_attr_to_inst`; `log_n_features`; `log_n_samples`; `max_log_cat_cardinality`; `mean_log_cat_cardinality`; `mean_numeric_unique_ratio`.
B	68	29	`duplicate_row_ratio`; `frac_bounded_01`; `frac_continuous_cols`; `frac_heavy_tail`; `frac_integer_valued`; `frac_multimodal`; `frac_nonneg`; `frac_outlier_rows`; `frac_skewed`; `hopkins`; `kmeans_silhouette_max`; `mean_n_modes`; `pymfe_attr_conc.mean`; `pymfe_attr_conc.sd`; `pymfe_attr_ent.mean`; `pymfe_attr_ent.sd`; `pymfe_cor.mean`; `pymfe_cor.sd`; `pymfe_frac_bin`; `pymfe_frac_norm`; `pymfe_frac_outliers`; `pymfe_kurtosis.mean`; `pymfe_kurtosis.sd`; `pymfe_nr_cor_attr`; `pymfe_skewness.mean`; `pymfe_skewness.sd`; `pymfe_sparsity.mean`; `pymfe_sparsity.sd`; `zero_ratio`.
C	16	16	`cat_num_association`; `frac_cols_with_missing`; `frac_rows_with_missing`; `frac_strong_cor_pairs`; `max_abs_cor`; `max_col_missing_ratio`; `max_norm_mutual_info`; `mean_abs_cor`; `mean_missing_per_missing_row`; `mean_norm_mutual_info`; `missing_cell_ratio`; `missing_mar_score`; `missing_pattern_count_log`; `missing_pattern_entropy`; `pca_dims_95_ratio`; `top_eigen_ratio`.
D	20	18	`class_entropy_norm`; `class_imbalance_log`; `is_binary`; `is_multiclass`; `is_regression`; `lm_best`; `lm_knn`; `lm_linear`; `lm_rf`; `lm_tree`; `log_n_classes`; `minority_class_ratio`; `n_targets_log`; `noise_level`; `nonlinearity_gain`; `target_skew_abs`; `target_unique_ratio`; `tree_vs_knn`.
Total	113	72	–

Sampling and failure handling. Dataset size, missingness rates, missingness patterns, and the duplicate-row ratio are computed on the full dataset. Other computationally expensive statistics use a random sample of at most 10,000 records. For meta-feature computation, we impute missing numerical values with the median and fill entirely missing numerical columns with 0. Missing categorical values are encoded as `__missing__`. These operations apply only to meta-feature computation and leave the original data unchanged. Failures are handled separately in the PyMFE, clustering-structure, dependency-structure, and landmarking modules. If a module fails, only its dimensions are marked as missing.

Landmarking. Following the landmarking principle (Pfahringer et al., 2000), we characterize target predictability with four fixed lightweight models: a logistic/ridge linear model, a depth-3 decision tree, 5-nearest neighbors, and a random forest with 100 trees (Hoerl & Kennard, 1970; Breiman et al., 1984; Cover & Hart, 1967; Breiman, 2001). Within each fold, numerical features are median-imputed and standardized. Categorical features are constant-imputed and one-hot encoded, with at most 20 categories retained per column. We remove records with missing targets and, for classification, classes with fewer than 6 samples. If more than 5,000 records remain, 5,000 are sampled without replacement using random seed 0, followed by three-fold cross-validation. This limit applies only to auxiliary meta-feature computation; the final dataset is not truncated. For a K-class classification task and a regression task, respectively, the scores are

$$s_{\text{cls}} = \text{clip}\left(\frac{\text{BalAcc} - 1/K}{1 - 1/K}, 0, 1\right), \qquad s_{\text{reg}} = \text{clip}(R^2, 0, 1). \tag{5}$$

These scores describe the dataset's target structure and exclude the performance of all methods evaluated in this paper.

A.5 Meta-Feature Cleaning and Normalization

Of the raw meta-features, 28 PyMFE statistics depend directly on the original measurement units, including means, standard deviations, extrema, covariances, and raw eigenvalues. Another 11 PyMFE counts or ratios duplicate the logarithmic or proportional representations in group A. The features `has_target` and `target_missing_ratio` are nearly constant in the current candidate pool.

Removing these 41 dimensions leaves 72 meta-features. We apply no further within-group correlation pruning. PCA summarizes the remaining correlated variation in a lower-dimensional representation.

For datasets without an identified target, missing group-D meta-features are imputed using the corresponding median over datasets with valid targets. All other missing values are imputed using the median of the corresponding dimension over the full candidate pool. Any meta-feature with more than 20% missing values would also be removed during preprocessing.

Meta-features with at most two observed unique values are standardized using a z-score. All other dimensions are mapped to a standard normal distribution by a `QuantileTransformer` (Pedregosa et al., 2011) fitted with at most 200 quantiles, then clipped to $[-3.5, 3.5]$:

$$\widetilde{h}_{ij} = \mathrm{clip}\Big[\Phi^{-1}\Big(\widehat{F}_j^{(200)}(h_{ij})\Big), -3.5, 3.5\Big], \tag{6}$$

where $\widehat{F}_j^{(200)}$ denotes the quantile map fitted with at most 200 empirical quantiles. Of the 72 retained dimensions, 69 use quantile normalization and 3 binary task-type indicators use z-score standardization.

A.6 PCA Representation, Clustering, and Representative Selection

PCA representation. Let $H \in \mathbb{R}^{377\times 72}$ denote the standardized candidate-dataset matrix. We fit PCA (Jolliffe, 2002) without whitening and choose the smallest dimension q satisfying

$$q = \min\left\{ r : \frac{\sum_{j=1}^{r} \lambda_j}{\sum_{j=1}^{72} \lambda_j} \geq 0.99 \right\}. \tag{7}$$

This gives $q = 41$ and 99.0726% cumulative explained variance. The 41 principal components (PCs) are linear combinations of the 72 meta-features, not a subset of 41 original features. Figure 7(b) plots the cumulative explained variance against the number of PCs and marks the 41-component cutoff required to reach the 99% threshold.

Clustering and representative datasets. Let $\mathbf{e}_i \in \mathbb{R}^{41}$ denote the PCA coordinates of dataset $\mathcal{D}_i$. We apply k-means (Lloyd, 1982) in this space with $k = 50$, `n_init`$= 10$, and random seed 0. The clustering objective is

$$\min_{\{a_i\},\{\boldsymbol{\mu}_c\}} \sum_{i=1}^{377} \|\mathbf{e}_i - \boldsymbol{\mu}_{a_i}\|_2^2 . \tag{8}$$

Clusters are processed in decreasing order of size. Within each cluster, we select the candidate nearest to the centroid among those allowed by the same-source rule. Let $\mathcal{E}_c$ denote the set of eligible nonduplicate candidates when cluster c is processed. The selected representative is

$$\mathcal{D}_c^{\star} = \arg\min_{\mathcal{D}_i \in C_c \cap \mathcal{E}_c} \|\mathbf{e}_i - \boldsymbol{\mu}_c\|_2 . \tag{9}$$

The representative is an observed candidate dataset; the synthetic centroid cannot be evaluated directly.

Same-source version suppression. Dataset names are converted to lowercase and stripped of nonalphanumeric characters. Two datasets are marked as potential versions of the same source if their normalized names are identical, or if the shorter normalized name has at least 5 characters and is a prefix of the other, provided that their sample counts differ by no more than a factor of 1.5. After one version is selected, other versions from the same source cannot represent later clusters. This procedure excludes 31 potential duplicates and selects one representative from each of the 50 clusters. Figure 8 shows a two-dimensional UMAP projection (McInnes et al., 2018) of the candidate datasets and selected representatives.

A.7 Final Benchmark, Data Splits, and Structural Evaluation Set

OpenML evaluation set. The 50 selected OpenML-hosted datasets comprise 26 binary-classification, 9 multiclass-classification, and 15 regression datasets. By source type, 40 datasets are

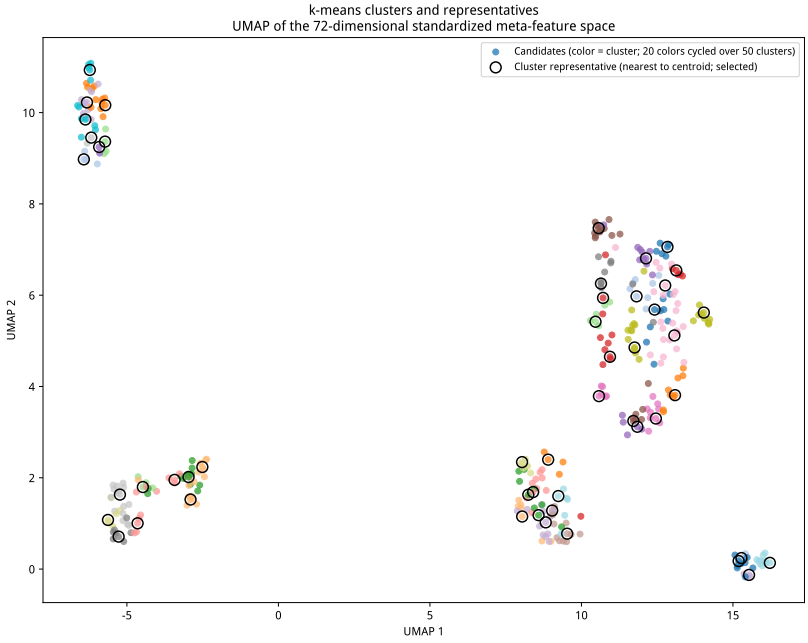


Figure 8: Two-dimensional UMAP projection of the candidate datasets and selected representatives. Colors denote k-means clusters, with the palette reused across clusters, and black circles mark the selected representative of each cluster. UMAP is used only for visualization; PCA, clustering, and representative selection are all performed in the high-dimensional space.

real-world or derived from real-world data, 9 are synthetic, simulated, or manually constructed, and 1 combines real and synthetic data. Task type and source type are separate classification schemes. The datasets contain 106–53,940 samples and 4–126 features. The largest table has 690,000 cells, and 8 datasets contain native missing values.

Fixed train–test split. Dataset selection is completed before the final train–test split. Let $\mathcal{D}_i = \{z_{ij}\}_{j=1}^{N_i}$ have sample-index set $I_i = \{1, \ldots, N_i\}$. We define:

$$\begin{aligned} &\mathcal{D}_{i,\text{train}} = \{z_{ij} : j \in I_{i,\text{train}}\}, && \mathcal{D}_{i,\text{test}} = \{z_{ij} : j \in I_{i,\text{test}}\}, \\ &I_{i,\text{train}} \cap I_{i,\text{test}} = \varnothing, && I_{i,\text{train}} \cup I_{i,\text{test}} = I_i. \end{aligned} \tag{10}$$

By default, the data are split once in an 80%/20% ratio. Classification datasets are stratified by the target whenever class counts permit. Generation, imputation, and prediction share the same fixed split. All model parameters, value domains, category sets, and normalization statistics are determined only from the training set. Methods that require validation data create an internal split from the training set, while the test set is reserved for final held-out evaluation.

Structural causal model evaluation set. In addition to the OpenML evaluation set, we include the six structural causal model datasets used by TabStruct (Jiang et al., 2026): Hailfinder, Healthcare, Insurance, Magic-Irri, Mehra, and Sangiovese. Their data-generating mechanisms or variable-dependence structures are specified in advance, allowing controlled evaluation of structural recovery. These datasets are excluded from the clustering of the OpenML candidate pool and are reported separately as the structural evaluation set.

A.8 Complete Dataset Inventory

Table 7 summarizes the complete 50-dataset OpenML evaluation set and the six-dataset structural evaluation set, including their task types, sample and feature counts, variable composition, and class statistics where applicable.

Table 7: Complete inventory of the 56 benchmark datasets. N combines the fixed training and test splits; D excludes the target. Num., Cat., and ID give the numbers of numerical, categorical, and identifier columns. Class counts and class-size ranges apply only to classification datasets.

Dataset	Source	OML ID	Task	N	D	N/D	Num.	Cat.	ID	Cls.	Class min	Class max
adult-census	OpenML	1119	Cls.	32,561	15	2170.73	6	8	1	2	7,841	24,720
allbp	OpenML	40707	Cls.	3,772	29	130.07	6	23	0	3	14	3,609
Amazon_employee_access	OpenML	46905	Cls.	32,769	9	3641.00	0	9	0	2	1,897	30,872
appendicitis_test	OpenML	45589	Cls.	106	7	15.14	7	0	0	2	21	85
Australian	OpenML	40981	Cls.	690	14	49.29	6	8	0	2	307	383
auto_price	OpenML	745	Cls.	159	15	10.60	14	1	0	2	54	105
avocado_sales	OpenML	43927	Reg.	18,249	13	1403.77	11	2	0	–	–	–
bank32nh	OpenML	833	Cls.	8,192	32	256.00	32	0	0	2	2,543	5,649
BNG-echoMonths	OpenML	1199	Reg.	17,496	9	1944.00	6	3	0	–	–	–
Brazilian_houses	OpenML	44062	Reg.	10,692	11	972.00	8	3	0	–	–	–
car_evaluation	OpenML	43921	Cls.	1,728	6	288.00	0	6	0	4	65	1,210
cleveland	OpenML	786	Cls.	303	13	23.31	6	7	0	2	139	164
colleges	OpenML	42727	Reg.	7,063	45	156.96	32	12	1	–	–	–
CPMP-2015-regression	OpenML	41700	Reg.	2,108	25	84.32	23	2	0	–	–	–
cpu_act	OpenML	45029	Reg.	8,192	21	390.10	21	0	0	–	–	–
credit_card_clients_default	OpenML	46919	Cls.	30,000	23	1304.35	20	3	0	2	6,636	23,364
delta_ailerons	OpenML	803	Cls.	7,129	5	1425.80	5	0	0	2	3,346	3,783
diamonds	OpenML	46923	Reg.	53,940	9	5993.33	6	3	0	–	–	–
E-CommereShippingData	OpenML	46924	Cls.	10,999	10	1099.90	6	4	0	2	4,436	6,563
FICO-HELOC-cleaned	OpenML	45554	Cls.	9,871	23	429.17	21	2	0	2	4,735	5,136
FOREX_audjpy-hour-High	OpenML	41718	Cls.	43,825	10	4382.50	10	0	0	2	21,254	22,571
healthcare_insurance_expenses	OpenML	46931	Reg.	1,338	6	223.00	3	3	0	–	–	–
heloc	OpenML	45026	Cls.	10,000	22	454.55	22	0	0	2	5,000	5,000
houses	OpenML	46934	Reg.	20,640	8	2580.00	8	0	0	–	–	–
JapaneseVowels	OpenML	976	Cls.	9,961	14	711.50	14	0	0	2	1,614	8,347
jungle_chess_2pcs_raw_endgame_complete	OpenML	41027	Cls.	44,819	6	7469.83	6	0	0	3	4,335	23,062
kc1	OpenML	1067	Cls.	2,109	21	100.43	21	0	0	2	326	1,783
KDDCup09_upselling	OpenML	44186	Cls.	5,128	49	104.65	34	15	0	2	2,564	2,564
kr-vs-kp	OpenML	3	Cls.	3,196	36	88.78	0	36	0	2	1,527	1,669
MiamiHousing2016	OpenML	43093	Reg.	13,932	16	870.75	16	0	0	–	–	–
MiceProtein	OpenML	40966	Cls.	1,080	81	13.33	77	3	1	8	105	150
mushroom	OpenML	43923	Cls.	12,960	8	1620.00	0	8	0	5	2	4,320
ozone-level-8hr	OpenML	1487	Cls.	2,534	72	35.19	72	0	0	2	160	2,374
pendigits	OpenML	42857	Cls.	10,992	16	687.00	16	0	0	9	6	3,566
Performance-Prediction	OpenML	43812	Cls.	1,340	19	70.53	19	0	0	2	509	831
planning-relax	OpenML	1490	Cls.	182	12	15.17	12	0	0	2	52	130
primary-tumor	OpenML	1003	Cls.	339	17	19.94	0	17	0	2	84	255
puma8NH	OpenML	816	Cls.	8,192	8	1024.00	8	0	0	2	4,078	4,114
pwLinear	OpenML	721	Cls.	200	10	20.00	10	0	0	2	97	103
qsar-biodeg	OpenML	1494	Cls.	1,055	41	25.73	41	0	0	2	356	699
QSAR_fish_toxicity	OpenML	46954	Reg.	907	6	151.17	6	0	0	–	–	–
SolarPower	OpenML	42328	Reg.	204	4	51.00	4	0	0	–	–	–
steel-plates-fault	OpenML	40982	Cls.	1,941	27	71.89	27	0	0	7	55	673
stock_fardamento02	OpenML	42545	Reg.	6,277	6	1046.17	5	1	0	–	–	–
us_crime	OpenML	42730	Reg.	1,994	126	15.83	126	0	0	–	–	–
vehicle	OpenML	994	Cls.	846	18	47.00	18	0	0	2	218	628
website_phishing	OpenML	46963	Cls.	1,353	9	150.33	0	9	0	3	103	702
wine	OpenML	44091	Cls.	2,554	11	232.18	11	0	0	2	1,277	1,277
wine_quality	OpenML	44136	Reg.	6,497	11	590.64	11	0	0	–	–	–
WineDataset	OpenML	43571	Cls.	178	13	13.69	13	0	0	3	48	71
hailfinder	TabStruct	–	Cls.	20,000	55	363.64	0	55	0	3	4,988	8,993
healthcare	TabStruct	–	Reg.	20,000	6	3333.33	3	3	0	–	–	–
insurance	TabStruct	–	Cls.	20,000	26	769.23	0	26	0	4	367	11,176
magic-irri	TabStruct	–	Reg.	20,000	63	317.46	63	0	0	–	–	–
mehra	TabStruct	–	Reg.	20,000	23	869.57	17	6	0	–	–	–
sangiovese	TabStruct	–	Cls.	20,000	14	1428.57	14	0	0	16	1,164	1,359

B METHODS AND QUALITY METRIC COLLECTION

This appendix lists the selected method identities and the quality metrics displayed in Figures 9 and 10. The collection contains 20 generation, 10 imputation, and 15 prediction metrics. The 20 retained semantic axes and their 42 metric members are specified separately in Table 3.

B.1 METHOD CAPABILITIES

Table 8 lists 45 selected method identities, including benchmark controls. G, I, and P denote generation, imputation, and prediction. Support refers to the implemented benchmark interface, including conditional or adaptation routes. Methods with the same configuration share one cell.

Table 8: Selected methods grouped by benchmark capability. Each method is cited separately; symbols indicate supported and unsupported interfaces.

Methods	G	I	P
ForestDiffusion-VP (joint) (Jolicoeur-Martineau et al., 2024); Gaussian copula (Patki et al., 2016); GReaT (Borisov et al., 2023); LimiX-2 (Zhang et al., 2026); TabDDPM (Kotelnikov et al., 2023); TabDiff (Shi et al., 2025); TabICLv2 (Qu et al., 2026); TabPFN-3 + unsupervised (Grinsztajn et al., 2026; Prior Labs, 2026); TabSyn (Zhang et al., 2024).	✓	✓	✓
ForestDiffusion-VP (label-conditioned) (Jolicoeur-Martineau et al., 2024).	✓	✓	×
ARF (Watson et al., 2023); Bayesian network (Qian et al., 2023); CTGAN (Xu et al., 2019); Independent marginals[‡]; NFlow (Durkan et al., 2019; Qian et al., 2023); NRGBoost (Bravo, 2025); SCM (DirectLiNGAM) (Shimizu et al., 2011); SMOTE (Chawla et al., 2002); SMOTE-NC (Chawla et al., 2002); SMOTER-style interpolation (Torgo et al., 2013); TabEBM (Margeloiu et al., 2024); Training-row bootstrap (Efron, 1979); TVAE (Xu et al., 2019).	✓	×	×
DiffPuter (Zhang et al., 2025a); HyperImpute (Jarrett et al., 2022); kNN imputation (Troyanskaya et al., 2001; Pedregosa et al., 2011); Mean/mode imputation (Pedregosa et al., 2011); Median/mode imputation (Pedregosa et al., 2011); MICE[†] (van Buuren & Groothuis-Oudshoorn, 2011); MissForest[†] (Stekhoven & Bühlmann, 2012); ReMasker (Du et al., 2024); TabCSDI (Zheng & Charoenphakdee, 2022); TabImpute (Feitelberg et al., 2025).	×	✓	×
CARD (Han et al., 2022); CatBoost (Prokhorenkova et al., 2018); FT-Transformer (Gorishniy et al., 2021); LightGBM (Ke et al., 2017); LightGBM-Quantile (Ke et al., 2017); Logistic regression (Pedregosa et al., 2011); MLP (Pedregosa et al., 2011); NGBoost (Duan et al., 2020); Random forest (Breiman, 2001); Ridge regression (Hoerl & Kennard, 1970; Pedregosa et al., 2011); TabM (Gorishniy et al., 2025); XGBoost (Chen & Guestrin, 2016).	×	×	✓

[†] MICE and MissForest denote the benchmark implementations. [‡] Independent marginals is a benchmark-defined control that samples each column independently from its empirical training marginal (`methods/independent_marginal/adapter.py`). The analysis alias map assigns historical TabICL outputs to the displayed TabICLv2 identity.

B.2 QUALITY METRIC COLLECTION

The tables summarize each metric. Each stored key consists of a family prefix and a leaf name. Arrows indicate the ranking preference: $\uparrow$ larger, $\downarrow$ smaller, and $\rightarrow a$ closer to a. Superscript D denotes a paired replicate diagnostic. Privacy diagnostics do not constitute privacy guarantees.

B.2.1 GENERATION

Metrics compare real and synthetic tables.

Table 9: Generation quality metric collection (20 metrics).

Metric (exact leaf)	Core meaning
generation.fidelity.* (SDV, 2026)	
categorical_tvd_error	↓ Mismatch in categorical marginal distributions.
mixed_association_error	↓ Mismatch in pairwise association strengths across mixed column types.
numerical_ks_error	↓ Mismatch in numerical marginal distributions.
pairwise_trend_error	↓ Mismatch in signed numerical trends and categorical joint distributions.
shape_error	↓ Overall numerical and categorical marginal mismatch.
generation.distinguishability.* (Lopez-Paz & Oquab, 2017)	
gbdt_c2st_auroc	→ 0.5 Real-versus-synthetic discrimination by gradient-boosted trees.
logistic_c2st_auroc	→ 0.5 Real-versus-synthetic discrimination by logistic regression.
generation.query.* (McKenna et al., 2019)	
one_way_error	↓ Real-versus-synthetic discrepancy in one-column query answers.
two_way_error	↓ Real-versus-synthetic discrepancy in two-column query answers.
three_way_error	↓ Real-versus-synthetic discrepancy in three-column query answers.
generation.support.* (Alaa et al., 2022)	
alpha_precision	↑ Fidelity of synthetic samples to real-data support.
authenticity	↑ Synthetic novelty under the benchmark's real-row-based proxy.
beta_recall_coverage	↑ Coverage of real-data support by synthetic samples.
generation.utility.* (Esteban et al., 2017)	
global_utility_ratio	↑ Synthetic-to-real predictive utility across column targets.
tstr_gap	→ 0 Target-prediction utility gap between synthetic-trained and real-trained models.
generation.memorization.*	
exact_copy_rate	↓ Frequency of exact training-record copies in synthetic data.
generation.privacy.* (Carlini et al., 2022)	
delta_presence	↓ Cluster-level training-versus-synthetic presence discrepancy; a noncanonical benchmark diagnostic.
distance_mia_auroc	→ 0.5 Distance-based discrimination of training members from held-out nonmembers.
distance_mia_tpr_at_1pct_fpr	↓ Membership-attack sensitivity at a 1% false-positive rate.
mean_min_euclidean_dcr	↑ Euclidean separation between synthetic records and their nearest training records.

B.2.2 IMPUTATION

Metrics describe masked-feature recovery in the target-observed setting.

Table 10: Imputation quality metric collection (10 metrics).

Metric (exact leaf)	Core meaning
imputation.point.*	
categorical_balanced_accuracy	↑ Class-balanced accuracy of recovered categorical values.

Continued on next page

Table 10 (continued)

Metric (exact leaf)	Core meaning
categorical_macro_f1	↑ Class-balanced precision and recall of recovered categorical values.
macro_nmae	↓ Normalized absolute error of numerical reconstruction.
macro_nrmse	↓ Normalized numerical reconstruction error, emphasizing larger errors.
imputation.structure.*	
association_matrix_error	↓ Relative distortion of pairwise dependence.
categorical_jsd	↓ Symmetric discrepancy in categorical value distributions.
forward_kl	↓ Directional mismatch from true to imputed value distributions.
numeric_wasserstein1	↓ Normalized discrepancy in numerical value distributions.
imputation.downstream.*	
utility_gap	→ 0 Change in downstream predictive utility after imputation.
imputation.robustness.*	
missingness_shift_degradation[D]	↓ Worst reconstruction degradation under changing missingness conditions.

B.2.3 PREDICTION

The collection distinguishes classification (10 metrics) from point regression (5 metrics).

Table 11: Prediction quality metric collection (15 metrics).

Metric (exact leaf)	Core meaning
prediction.classification.*	
accuracy	↑ Fraction of correctly classified examples.
average_precision	↑ Precision-recall quality of probability-based class rankings.
balanced_accuracy	↑ Classification accuracy weighted equally across classes.
macro_f1	↑ Balance of precision and recall, weighted equally across classes.
mcc	↑ Correlation between predicted and true class labels.
roc_auc	↑ Class discrimination across decision thresholds.
prediction.calibration.* (Geifman et al., 2019; Gneiting & Raftery, 2007; Guo et al., 2017)	
aurc	↓ Prediction error as low-confidence cases are progressively withheld.
brier_score	↓ Squared discrepancy between class probabilities and observed outcomes.
ece	↓ Mismatch between predictive confidence and observed accuracy.
log_loss	↓ Probabilistic prediction error that penalizes confidently incorrect predictions.
prediction.regression.*	
mae	↓ Average absolute prediction error in target units.
median_absolute_error	↓ Typical absolute prediction error, less sensitive to outliers.
nrmse	↓ Prediction error relative to training-target variability.
r2	↑ Prediction quality relative to a test-mean baseline.
rmse	↓ Prediction error in target units, emphasizing large residuals.

C METRIC-SPACE AUDIT: PROTOCOL AND RESULTS

Section 3 defines the common-support rank protocol and the criteria for retaining capability axes. Table 3 in the main text specifies their exact metric membership. This appendix describes the audit protocol and task-specific evidence for these choices. Appendix B.2 defines the metrics displayed in the two rank-correlation figures.

For both generation and imputation, the mixed-type association matrix uses $|r|$ for numerical–numerical pairs, Cramér's V for categorical–categorical pairs (Cramér, 1946), and the correlation ratio η for mixed pairs (Pearson, 1905). The two tasks use different scalar summaries of this shared representation: generation reports the mean absolute difference over comparable off-diagonal pairs and its complement, whereas imputation reports the relative Frobenius distance. The metrics compare the same pairwise dependence representation but are not numerically interchangeable.

C.1 AUDIT PROTOCOL: COMMON-SUPPORT RANKS BEFORE CORRELATIONS

After constructing the ranks in Equation 1, the pairwise audit retains only cells in which both quantities are observed, discards strata in which either metric is constant, requires at least three methods per stratum and at least 30 paired cells over six datasets, and computes the Pearson correlation of the pooled rank pairs (the Spearman-equivalent construction used by the audit). The catalog sets $a_k = 0.5$ for C2ST and distance-MIA AUROC and $a_k = 0$ for utility gaps. The catalog marks normalized median Gower DCR as neutral; analyses that require a direction use the explicit provisional target-one transform $-|x - 1|$ and do not promote it to confirmatory privacy evidence. For the spectral audit, c is the fixed metric cohort. The strata are dataset for generation, dataset–missingness setting–rate–contract for imputation, and dataset–task type for prediction. Confidence intervals use 500 bootstrap resamples (Efron, 1979) with whole datasets as the resampling units.

For relationships with finite confidence intervals, we call a pair a *redundancy candidate* only when its oriented correlation satisfies $\rho \geq 0.85$ and the 95% dataset-bootstrap lower bound is at least 0.70. Conversely, $\rho \leq -0.50$ with an upper bound at most -0.20 marks a stable trade-off. These thresholds inform, but do not replace, semantic judgment. For the cluster counts in Table 2, we apply average linkage to $1 - |\rho|$ and cut at 0.15. This absolute-correlation clustering detects both duplicate orderings and two-sided trade-offs; only the former can be pruned. The heatmap ordering instead uses signed distance $1 - \rho$ so that opposing directions remain visually separate. The orderings in Figures 9 and 10 display the empirical blocks; adjacency alone does not justify merging metrics.

Figures 9 and 10 show the larger generation metric space separately from imputation and prediction to keep individual rank correlations legible.

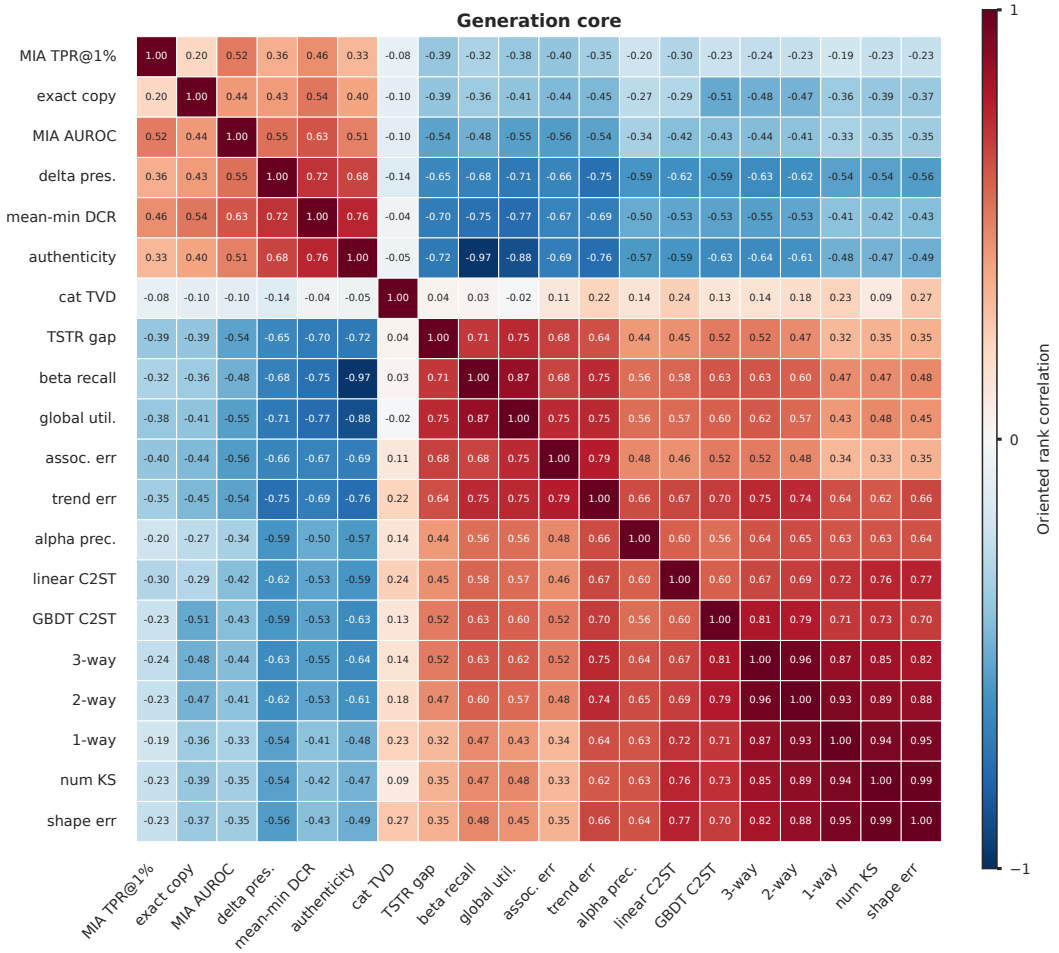


Figure 9: **Generation rank-correlation geometry (main metric-audit view).** Each cell reports the Pearson correlation ρ of common-support, within-dataset percentile ranks after orienting values so that larger is better, rounded to two decimals. Red denotes aligned method orderings and blue denotes trade-offs; average-linkage ordering uses distance $1 - \rho$. The map shows a dense marginal/query-fidelity block, while utility and β-recall align with one another but oppose authenticity and distance-to-training diagnostics. White denotes correlation near zero.

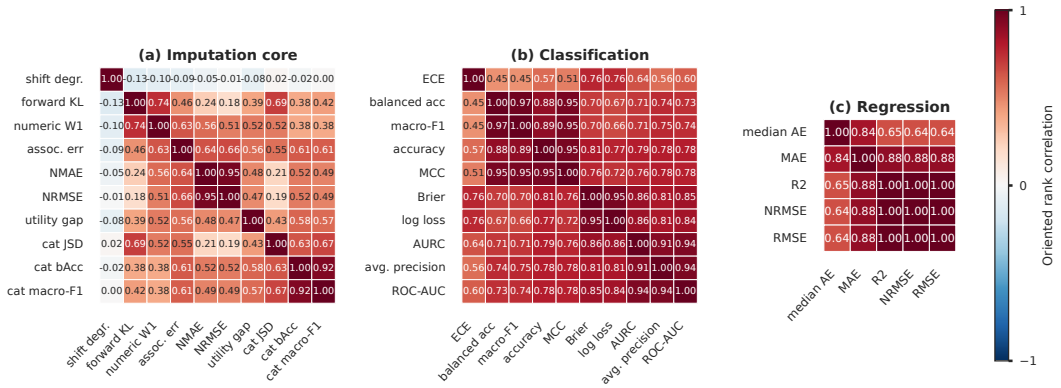


Figure 10: **Rank-correlation geometry for imputation and prediction.** Cell values report oriented rank correlation ρ, rounded to two decimals, under the same common-support construction and color scale as Figure 9. Imputation contains tight numerical- and categorical-point-accuracy pairs but separates distributional, association, and utility quantities. Classification retains distinct hard-decision, proper-score, ranking/selective-risk, and calibration views despite broad shared signal. Regression exhibits an almost rank-equivalent squared-error block while retaining distinct absolute-error summaries.

C.2 From correlation blocks to redundancy clusters

Figures 9 and 10 inform metric selection. Within each task, a positive block supports pooling only when its metrics share an estimand; weakly aligned metrics are not pruned and can share a named construct only as reportable complementary probes; mixed-sign blocks are retained as explicit trade-offs. The following decisions account for each pooled family and retained direction in Table 3.

Generation. The full 29-metric quality audit yields 17 local correlation clusters, the largest number among the three tasks. Figure 9 shows the fixed 20-metric core; the full audit additionally evaluates alternate parameterizations and validity guards. The five marginal/query-fidelity metrics form the clearest positive block: pairwise correlations range from 0.823 to 0.990, and eight of ten pairs pass the conservative redundancy rule. An equivalent shape metric brings the cluster to six. By contrast, the two C2ST variants are only moderately aligned ($\rho = 0.599$), so linear and nonlinear distinguishability remain separate, reportable probes within the named C2ST axis rather than statistical substitutes. We pool the dense marginal/query block into *fidelity*, but keep α-precision, association recovery, and TSTR separate because support, dependence, and transfer are not interchangeable with marginal fidelity. Four removable generation metrics are exact alternative parameterizations; semantic review identifies only five additional metrics as substantive reductions.

A *signed* cluster requires separate treatment. Global utility and β-recall align ($\rho = 0.872$), whereas authenticity opposes both global utility ($\rho = -0.881$) and β-recall ($\rho = -0.973$). Absolute correlation therefore places them nearby, but they measure opposite ends of a support–novelty trade-off and each must remain reportable. The privacy metrics also resist reduction: mean-min Euclidean DCR has correlations of 0.464, 0.630, and 0.720 with MIA TPR@1%, MIA AUROC, and delta presence. We group these as *privacy-risk diagnostics* because they provide complementary views of record proximity, membership inference, and group-presence disclosure, rather than estimates of one interchangeable quantity. None is a privacy guarantee, and exact-copy rate remains a separate memorization guardrail. These decisions yield the seven generation axes in Table 3.

Imputation. The imputation panel of Figure 10 contains ten quality metrics in eight clusters, six of them singletons. The only stable substitutions are numerical NMAE–NRMSE ($\rho = 0.947$, 95% CI $[0.931, 0.959]$) and categorical balanced accuracy–macro-F1 ($\rho = 0.925$, $[0.899, 0.949]$). Point accuracy therefore does not subsume distribution matching, association recovery, or downstream utility. Missingness-shift degradation is nearly orthogonal to all level metrics, but constant imputers can appear artificially robust because their overall performance is already at a floor; we retain it as a paired diagnostic rather than a standalone ranking axis. Draw-level quantities likewise remain

diagnostic rather than entering the reduced headline registry. Accordingly, Table 3 averages only the two point-recovery pairs and retains distribution, association, and utility as separate headline axes.

Prediction. The prediction panels in Figure 10 show four observed clusters among the ten classification metrics. Hard-decision correlations range from 0.882 to 0.971; Brier score and log loss correlate at 0.951; and correlations among ROC–AUC, average precision, and AURC range from 0.909 to 0.943. Accordingly, Table 3 pools hard decisions, proper scores, and ranking/selective risk within their respective axes, while ECE remains a singleton because calibration is not equivalent to discrimination, proper scoring, or selective prediction. In regression, R^2, NRMSE, and RMSE induce almost identical method orderings ($\rho > 0.999$), MAE joins this block at the 0.85 cut, and median absolute error remains separate. We nevertheless report MAE independently because mean absolute and squared error encode different loss functionals. Cluster membership indicates similar rankings but does not make the estimands interchangeable. This yields four classification and three regression axes in Table 3.

C.3 Construct-specific stress tests

Controlled synthetic examples illustrate how selected metrics respond to specific perturbations.[1] Independently shuffling columns preserves their marginal distributions while disrupting associations; replaying training rows can improve apparent fidelity while worsening memorization diagnostics; marginal-only imputation can approximate univariate distributions without accurately recovering individual entries or joint structure; and monotone probability rescaling can alter calibration without changing ranking. In these examples, marginal fidelity, dependence, memorization, point recovery, and calibration metrics respond differently to the imposed perturbations.

C.4 Reliability qualifies every redundancy claim

Measurement noise can obscure correlation structure or create apparent structure. Figure 11 therefore reports dataset- blocked rank ICC (Shrout & Fleiss, 1979) alongside two operational quantities. The separated-pair fraction counts method pairs whose replicate-mean difference exceeds 1.96 times the seed-level standard error of the difference; this threshold describes resolving power and is not a significance test. Top-method agreement is the mean Jaccard overlap of the best-method sets across replicate pairs. These summaries measure different properties: even a highly repeatable metric may identify different winners when several methods perform similarly. Among selected direction-defined generation components, rank ICCs range from 0.592 to 0.963. MIA TPR@1% and MIA AUROC form the low end at 0.592 and 0.674, and winner agreement can be as low as 0.161. These values support retaining component-level privacy diagnostics alongside their summary. For classification, correcting correlations by the three-seed test–retest reliabilities merges hard-decision, proper-score, and ranking metrics at the same 0.85 cut, while ECE remains separate. We treat this correction as a sensitivity analysis rather than a mechanical selection rule: it supports one broad predictive-quality factor plus a distinct calibration direction.

[1] The script and result tables are included in the supplementary material under `code/tabtoolkit_v1/outputs/analysis_metric/`: `metric_stress_tests.py`, `tables/metric_stress_test_raw.csv`, and `tables/metric_stress_test_oriented.csv`.

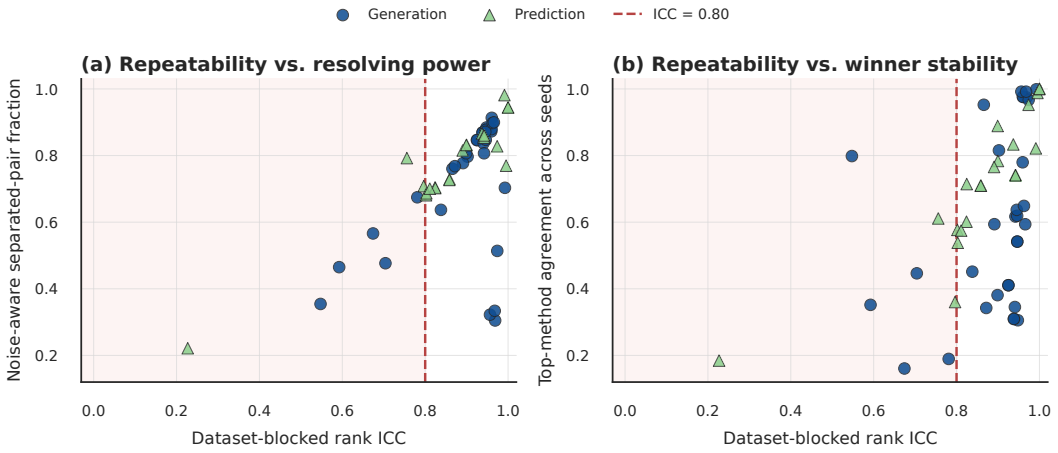


Figure 11: **Reliability and operational discrimination (auxiliary metric-audit view).** Each point is a direction-defined, informative catalog metric from generation or prediction with at least 20 repeated cells across at least eight datasets. The dashed line is a visual reference at rank ICC $= 0.80$; the shaded region lies below it. Left: the fraction of repeated method pairs separated by more than 1.96 times the seed-level standard error of their difference. Right: agreement of top-method sets across replicate pairs. Repeatability, pairwise resolving power, and winner stability are related but non-equivalent properties.

C.5 Compressed, but not one-dimensional

The PCA audit (Jolliffe, 2002) uses each task's full quality cohort. The Kaiser rule (Kaiser, 1960) retains five generation components (80% cumulative variance; PC1 53%), three imputation components (74%; 41%), two classification components (88%; 75%), and one regression component (86%; 86%). These counts are smaller than the corresponding $17/8/4/2$ local clusters because PCA captures shared variation rather than pairwise substitutability. A dominant quality factor can coexist with a scientifically important low-variance direction such as calibration or typical absolute error.

Figure 12 shows a complementary spectrum for the fixed core cohorts used in the heatmaps. The participation-ratio effective rank is 3.15 for the ten imputation-core metrics and 2.87 for the twenty generation-core metrics; reaching 90% spectrum mass requires five and eight components, respectively. Classification and point regression are more compressed (1.55 and 1.28), consistent with the tight blocks in Figure 10. These values describe a correlation spectrum, not performance. Because pairwise correlations can use different common-support cells, the assembled matrix need not be positive semidefinite; negative eigenvalues are clipped only for this descriptive spectrum. Compression does not imply one-dimensionality: the retained semantic axes preserve stable trade-offs and distinct estimands that a purely spectral summary would blend.

Using the retained axis counts $(7, 6, 4, 3)$ only as dimensionality budgets, the corresponding leading eigenspaces retain 88.22%, 95.46%, 96.49%, and 99.99% of the nonnegative spectrum for generation, imputation, classification, and regression. These are optimal same-dimensional spectral upper bounds, not the mass captured by the particular semantic composites. For the latter, let R_+ denote the nonnegative-eigenvalue reconstruction of the correlation matrix and let W encode the equal-weight composite members. We report the reconstruction fraction

$$\mathcal{C}(W) = \frac{\operatorname{tr}\left[R_+ W (W^\top R_+ W)^+ W^\top R_+\right]}{\operatorname{tr}(R_+)}, \tag{11}$$

where $(\cdot)^+$ is the Moore–Penrose inverse. The final composite spans retain 79.65%, 86.21%, 96.40%, and 99.99%, respectively. The generation value is necessarily partial: normalized median Gower DCR is neutral and therefore absent from R_+, so only the direction-defined members of the privacy-risk diagnostics axis enter this projection.

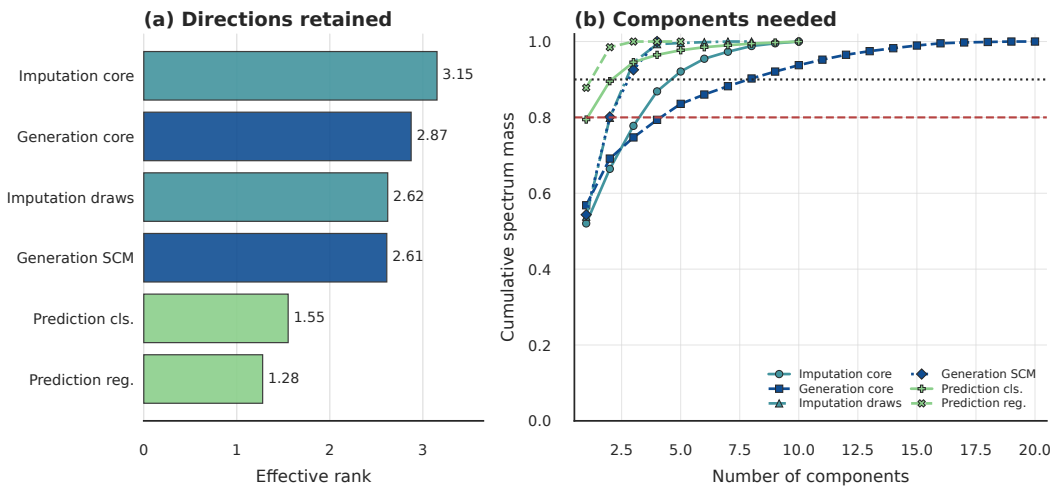


Figure 12: **Residual effective dimensionality (auxiliary metric-audit view).** Left: participation-ratio effective rank, $(\sum_j \lambda_j)^2 / \sum_j \lambda_j^2$, using the nonnegative-clipped correlation spectrum. Right: cumulative spectrum mass, with 80% and 90% guides. A cohort is shown only when every metric pair satisfies the common-support criteria stated above. "Imputation core" and "Imputation draws" use the target-observed contract; feature-only imputation is analyzed separately. The eight-metric draws spectrum summarizes additional distributional metrics. Generation SCM reports four structural/association quantities for the six SCM datasets, separately from the twenty-metric generation core. The curves summarize within-cohort compression.

D TASK RANK ANALYSIS DETAILS

This appendix details the procedure and aligned support for the cross-task rank analysis in Section 4. We compare method rankings across generation (G), imputation (I), classification prediction (P-cls), and regression prediction (P-reg). For each task pair, we restrict the analysis to the common method pool and recompute within-dataset ranks on the aligned support. This controls for differences in method availability, dataset difficulty, and metric scale.

D.1 TASK SETTINGS AND SHARED SUPPORT

Input units. The input table for each task is indexed by (d, m), where d denotes a dataset and m denotes a method. Each column contains one evaluation metric for that task. Raw records are first grouped using the declared seed/replicate fields and then averaged after metric orientation. Prediction datasets are separated into P-cls and P-reg according to task type. For imputation, we use the target-observed MCAR setting with a missing rate of 0.3.

Metric direction. We orient metrics with a clear optimization direction so that higher values indicate better performance. We negate metrics for which lower values are better, use $-|x|$ for metrics whose ideal value is 0, and use $-|x - 0.5|$ for metrics whose ideal value is 0.5. Neutral diagnostics retain their stored direction. For these diagnostics, a correlation sign describes the relation to the stored direction; it does not by itself indicate better or worse performance.

Common method pool. For tasks a and b, the common method pool is defined as $\mathcal{M}_{ab} = \mathcal{M}_a \cap \mathcal{M}_b$. Classification and regression use the same candidate intersections for I–P and G–P, but are analyzed independently on their respective dataset subsets. Table 12 reports the task-family candidate pools. We align support separately for each axis pair, retaining only dataset–method units with valid values for all declared members on both sides.

Table 12: Candidate method pools and member-complete aligned support for each task pair. "Datasets" and n give the axis-pair ranges of effective datasets and valid $(\text{dataset}, \text{method})$ units, respectively.

Task pair	Candidate pool	Datasets	Range of n	Candidate methods
I × P-cls	9	13–38	92–287	Forest-VP Joint, Gaussian Copula, GReaT, LimiX-2, TabDDPM, TabDiff, TabICLv2, TabPFN-3, TabSyn
I × P-reg	9	6–18	44–139	Forest-VP Joint, Gaussian Copula, GReaT, LimiX-2, TabDDPM, TabDiff, TabICLv2, TabPFN-3, TabSyn
G × I	10	12–56	98–482	Forest-VP, Forest-VP Joint, Gaussian Copula, GReaT, LimiX-2, TabDDPM, TabDiff, TabICLv2, TabPFN-3, TabSyn
G × P-cls	9	30–38	246–300	Forest-VP Joint, Gaussian Copula, GReaT, LimiX-2, TabDDPM, TabDiff, TabICLv2, TabPFN-3, TabSyn
G × P-reg	9	18	138–141	Forest-VP Joint, Gaussian Copula, GReaT, LimiX-2, TabDDPM, TabDiff, TabICLv2, TabPFN-3, TabSyn

D.2 COMPUTING AXIS-PAIR CORRELATIONS

Cell-wise alignment. For capability axes (c, c') from tasks (a, b), let $\mathcal{K}_c$ and $\mathcal{K}_{c'}$ denote their fixed member-metric sets. We define the aligned support for this axis pair as

$$\mathcal{S}^{ab}_{cc'} = \left\{ (d, m) : m \in \mathcal{M}_{ab}, \bigwedge_{k \in \mathcal{K}_c} v^{(a)}_{dmk} \text{ is valid}, \bigwedge_{\ell \in \mathcal{K}_{c'}} v^{(b)}_{dm\ell} \text{ is valid} \right\}. \tag{12}$$

Thus, a (d, m) unit is retained only when all declared members are valid on both sides. We construct $\mathcal{S}^{ab}_{cc'}$ independently for every axis pair rather than imposing complete support across all retained axes.

Within-dataset percentile ranks. Let $\mathcal{M}^{ab}_{d,cc'} = \{m : (d,m) \in \mathcal{S}^{ab}_{cc'}\}$. Every member metric on both sides is ranked over this same method set:

$$r^{(t)}_{d,m,k} = \frac{\mathrm{rank}_{\mathrm{avg}}\Big(v^{(t)}_{d,m,k}; \{v^{(t)}_{d,m',k} : m' \in \mathcal{M}^{ab}_{d,cc'}\}\Big)}{|\mathcal{M}^{ab}_{d,cc'}|} \in (0,1]. \tag{13}$$

Methods with identical metric values receive the average of the ranks they occupy. Percentile ranks are normalized by the size of the aligned method set and therefore lie in $(0,1]$, with the smallest possible value equal to $1/|\mathcal{M}^{ab}_{d,cc'}|$.

Axis aggregation. Following Equation 1, the axis score is the equal-weight mean of the declared member percentile ranks:

$$a^{(t)}_{d,m,c} = \frac{1}{|\mathcal{K}_c|} \sum_{k \in \mathcal{K}_c} r^{(t)}_{d,m,k}. \tag{14}$$

The member set is fixed at $\mathcal{K}_c$, and no additional standardization is applied after ranking. Member-complete alignment ensures that each axis score includes all its declared members.

Pooled Spearman correlation. After constructing the axis scores, we compute

$$\rho^{ab}_{cc'} = \mathrm{Spearman}\left(\{a^{(a)}_{d,m,c} : (d,m) \in \mathcal{S}^{ab}_{cc'}\}, \{a^{(b)}_{d,m,c'} : (d,m) \in \mathcal{S}^{ab}_{cc'}\}\right) \tag{15}$$

over all units in $\mathcal{S}^{ab}_{cc'}$. This is a single correlation over the pooled dataset–method units; we do not first compute a correlation within each dataset and then average correlations across datasets. A correlation is left undefined when fewer than 10 paired units remain.

D.3 Leave-One-Method-Out Influence Diagnostics

For every method $u \in \mathcal{M}_{ab}$, define the reduced method pool $\mathcal{M}^{(-u)}_{ab} = \mathcal{M}_{ab} \setminus \{u\}$. After each removal, we recompute the aligned support, within-dataset percentile ranks, axis scores, and final correlation $\rho^{ab,(-u)}_{cc'}$. A deletion is a no-op when it changes no valid unit for that axis pair. Let $\mathcal{E}^{ab}_{cc'}$ contain the effective deletions with finite recomputations. The LOMO sensitivity range and span are

$$\mathcal{R}^{ab}_{cc'} = \left[\min_{u \in \mathcal{E}^{ab}_{cc'}} \rho^{ab,(-u)}_{cc'}, \max_{u \in \mathcal{E}^{ab}_{cc'}} \rho^{ab,(-u)}_{cc'}\right], \qquad \Delta^{ab}_{cc'} = \max_{u \in \mathcal{E}^{ab}_{cc'}} \rho^{ab,(-u)}_{cc'} - \min_{u \in \mathcal{E}^{ab}_{cc'}} \rho^{ab,(-u)}_{cc'}. \tag{16}$$

An axis pair is called LOMO-direction-preserving when every effective recomputation has the same sign as its full-pool correlation. No-op deletions are excluded from the sign-agreement denominator and the effective deletion count. A dagger (†) marks axis pairs for which an effective deletion changes sign or produces zero correlation. The LOMO range is an influence diagnostic for the composition of the method pool, not a confidence interval or a significance test.

D.4 Cross-Task Axis-Level Correlations and LOMO Stability

Imputation and classification prediction (I $\times$ P-cls). Among the 24 axis pairs, 13 are LOMO-direction-preserving. Numerical point accuracy, association, and downstream utility gap have positive, direction-preserving correlations with all four classification axes. Categorical point accuracy has this relation only with hard-decision performance; neither distribution axis has a direction-preserving classification relation.

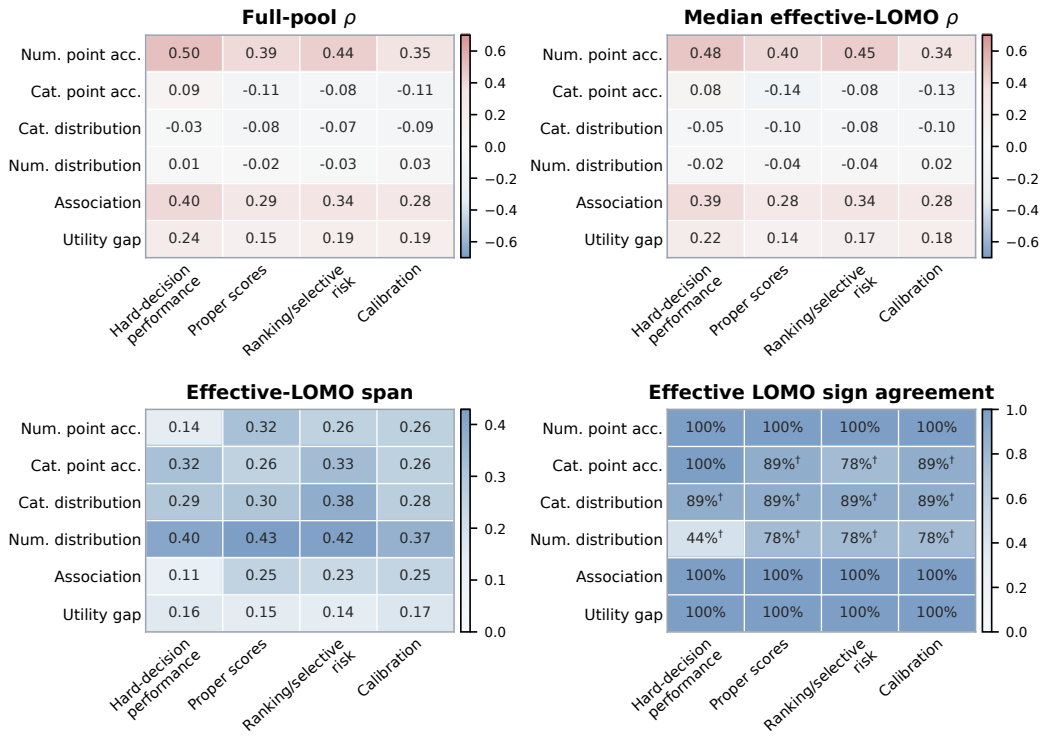


Figure 13: Axis-pair LOMO diagnostics for I × P-cls. From upper left to lower right, the panels show the pooled Spearman correlation for the full method pool, the median correlation across valid, effective single-method deletions, the LOMO span, and the fraction of effective deletions preserving the full-pool sign. Red and blue denote positive and negative correlations; sequential blue encodes the latter two diagnostics. A dagger (†) marks axis pairs for which at least one effective deletion changes sign or produces zero correlation. Each deletion requires realigning support, reranking within datasets, and recomputing axis scores; no-op deletions are excluded from the effective denominator. LOMO ranges are sensitivity diagnostics, not confidence intervals.

Imputation and regression prediction (I × P-reg). Fourteen of the 18 axis pairs are LOMO-direction-preserving. Numerical point accuracy, numerical distribution, association, and downstream utility gap have positive, direction-preserving correlations with all three regression axes. Categorical point accuracy also preserves a positive direction with squared error and mean absolute error, but not median error; categorical distribution is direction-unstable for all three.

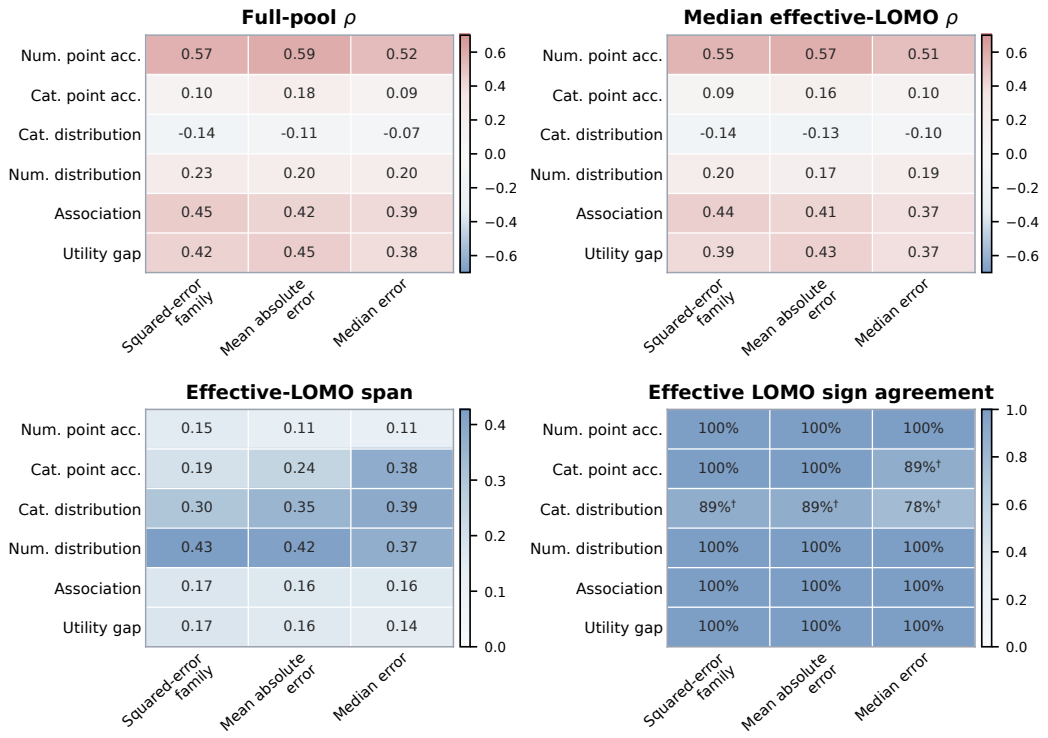


Figure 14: Axis-pair LOMO diagnostics for I × P-reg. From upper left to lower right, the panels show the pooled Spearman correlation for the full method pool, the median correlation across valid, effective single-method deletions, the LOMO span, and the fraction of effective deletions preserving the full-pool sign. Red and blue denote positive and negative correlations; sequential blue encodes the latter two diagnostics. A dagger (†) marks axis pairs for which at least one effective deletion changes sign or produces zero correlation. Each deletion requires realigning support, reranking within datasets, and recomputing axis scores; no-op deletions are excluded from the effective denominator. LOMO ranges are sensitivity diagnostics, not confidence intervals.

Generation and imputation (G × I). Of the 42 G × I pairs, 36 are LOMO-direction-preserving. TSTR gap and global-utility compromise are positive for all six imputation axes. Privacy-risk diagnostics are negative for five axes; numerical distribution is the exception. Fidelity preserves direction for five pairs, C2ST for four, and both α-precision and association for five; their signs remain axis-specific.

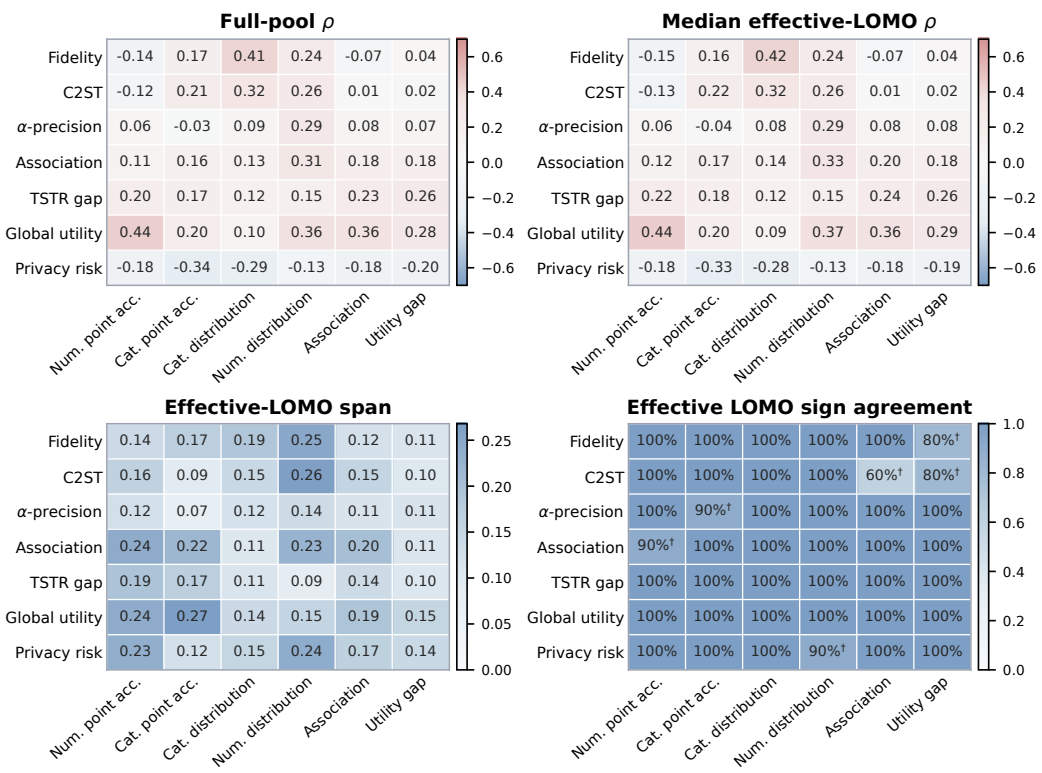


Figure 15: Axis-pair LOMO diagnostics for G × I. From upper left to lower right, the panels show the pooled Spearman correlation for the full method pool, the median correlation across valid, effective single-method deletions, the LOMO span, and the fraction of effective deletions preserving the full-pool sign. Red and blue denote positive and negative correlations; sequential blue encodes the latter two diagnostics. A dagger (†) marks axis pairs for which at least one effective deletion changes sign or produces zero correlation. Each deletion requires realigning support, reranking within datasets, and recomputing axis scores; no-op deletions are excluded from the effective denominator. LOMO ranges are sensitivity diagnostics, not confidence intervals.

Generation and classification prediction (G × P-cls). Seventeen of the 28 pairs are LOMO-direction-preserving. Fidelity and C2ST are negative with all four classification axes, while TSTR gap and global-utility compromise are positive with all four. Privacy-risk diagnostics also have positive, direction-preserving correlations with proper scores. All α-precision and association pairs are direction-unstable.

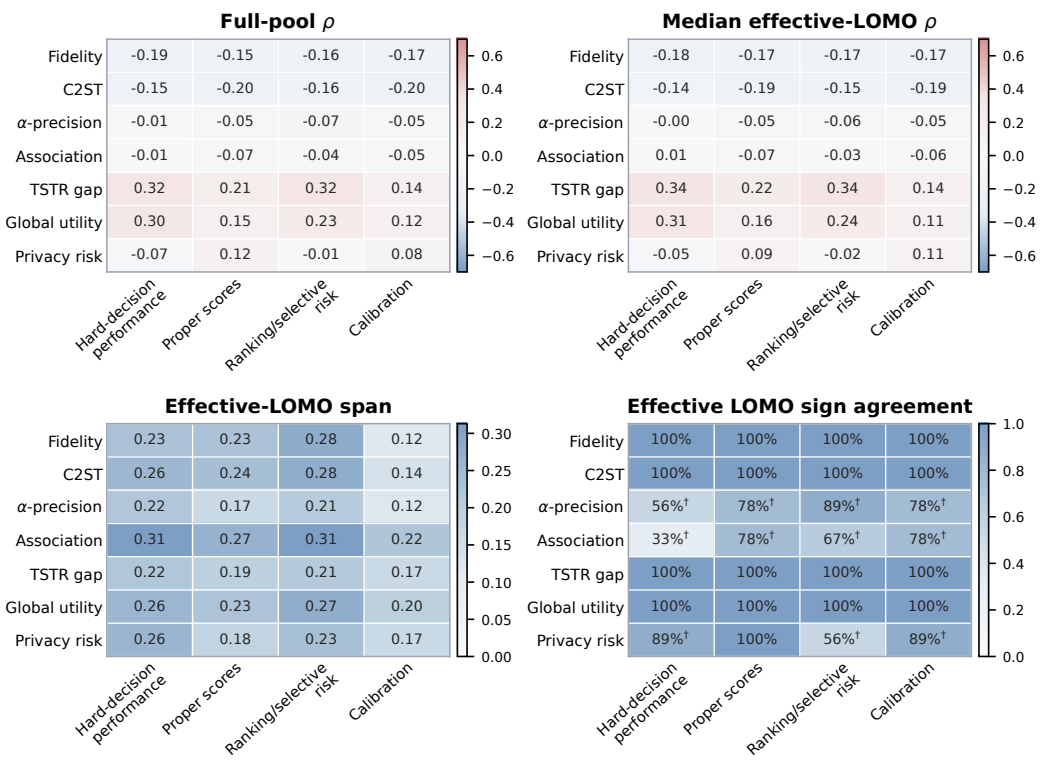


Figure 16: Axis-pair LOMO diagnostics for G × P-cls. From upper left to lower right, the panels show the pooled Spearman correlation for the full method pool, the median correlation across valid, effective single-method deletions, the LOMO span, and the fraction of effective deletions preserving the full-pool sign. Red and blue denote positive and negative correlations; sequential blue encodes the latter two diagnostics. A dagger (†) marks axis pairs for which at least one effective deletion changes sign or produces zero correlation. Each deletion requires realigning support, reranking within datasets, and recomputing axis scores; no-op deletions are excluded from the effective denominator. LOMO ranges are sensitivity diagnostics, not confidence intervals.

Generation and regression prediction (G × P-reg). Nine of the 21 pairs are LOMO-direction-preserving. TSTR gap and global-utility compromise are positive with all three regression axes. Fidelity and C2ST are negative only against squared error, while α-precision is positive only against median error. Every association and privacy-risk pair is direction-unstable.

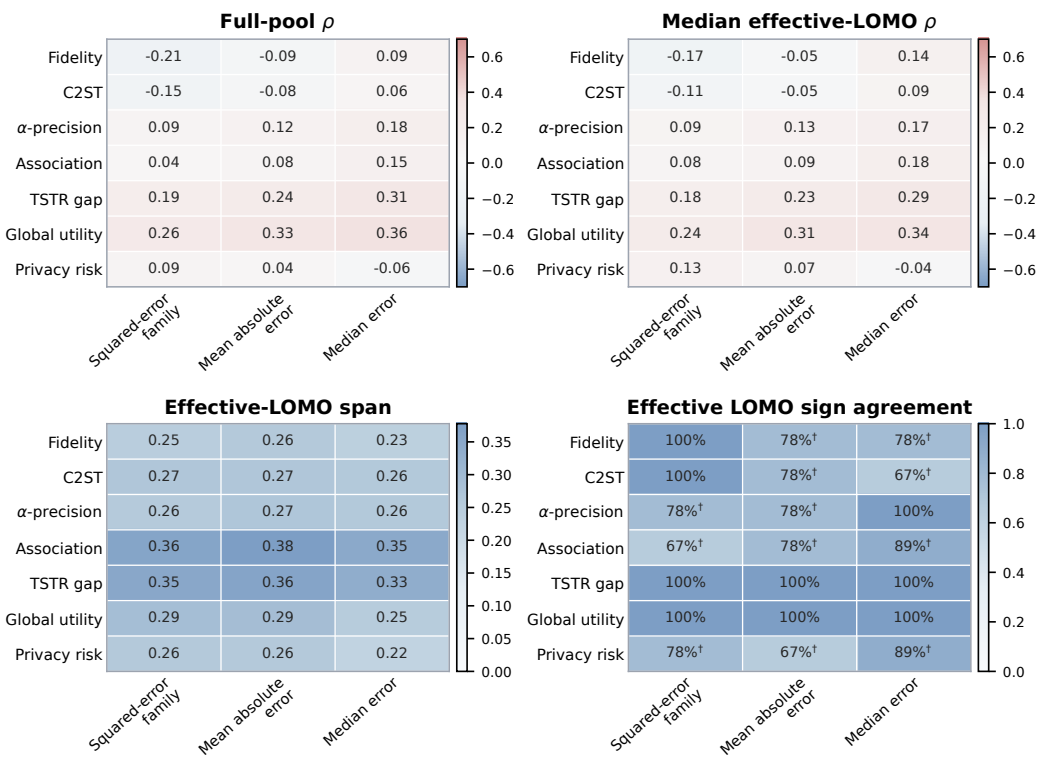


Figure 17: Axis-pair LOMO diagnostics for G × P-reg. From upper left to lower right, the panels show the pooled Spearman correlation for the full method pool, the median correlation across valid, effective single-method deletions, the LOMO span, and the fraction of effective deletions preserving the full-pool sign. Red and blue denote positive and negative correlations; sequential blue encodes the latter two diagnostics. A dagger (†) marks axis pairs for which at least one effective deletion changes sign or produces zero correlation. Each deletion requires realigning support, reranking within datasets, and recomputing axis scores; no-op deletions are excluded from the effective denominator. LOMO ranges are sensitivity diagnostics, not confidence intervals.

D.5 Complete Leave-One-Method-Out Grids

Figures 18–27 show the complete correlation grids for the full candidate pool and every LOMO recomputation. Each task pair occupies two pages for legibility. Each cell contains the pooled correlation coefficient. The grids identify which method removals account for changes in direction or large LOMO spans.

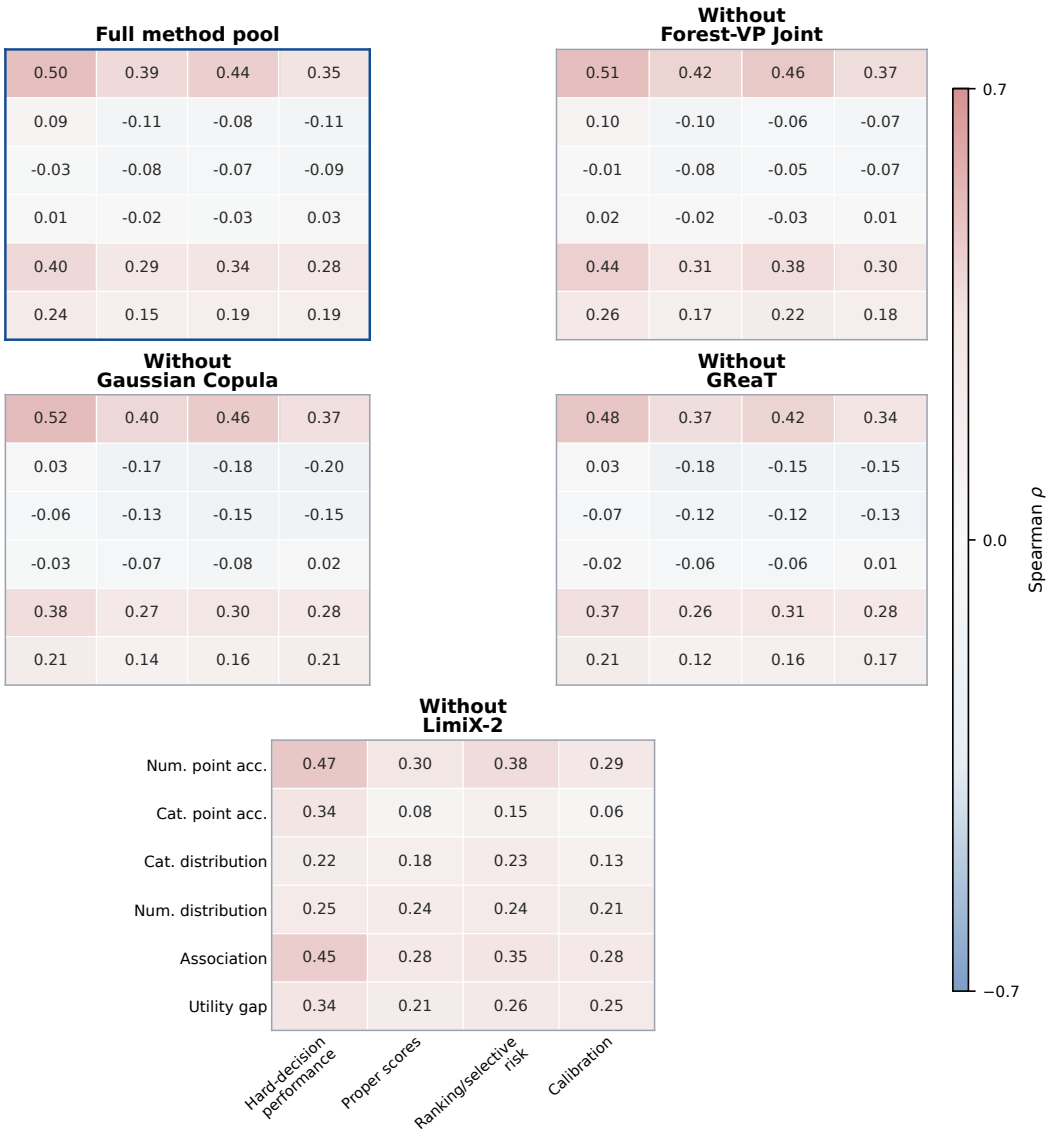


Figure 18: Complete leave-one-method-out results for I × P-cls (part 1 of 2). The "Full method pool" panel is the baseline; every other panel removes the named method. Each cell reports the pooled Spearman correlation. Red and blue denote positive and negative correlations. Support alignment, within-dataset ranks, and capability-axis scores are recomputed for every panel.

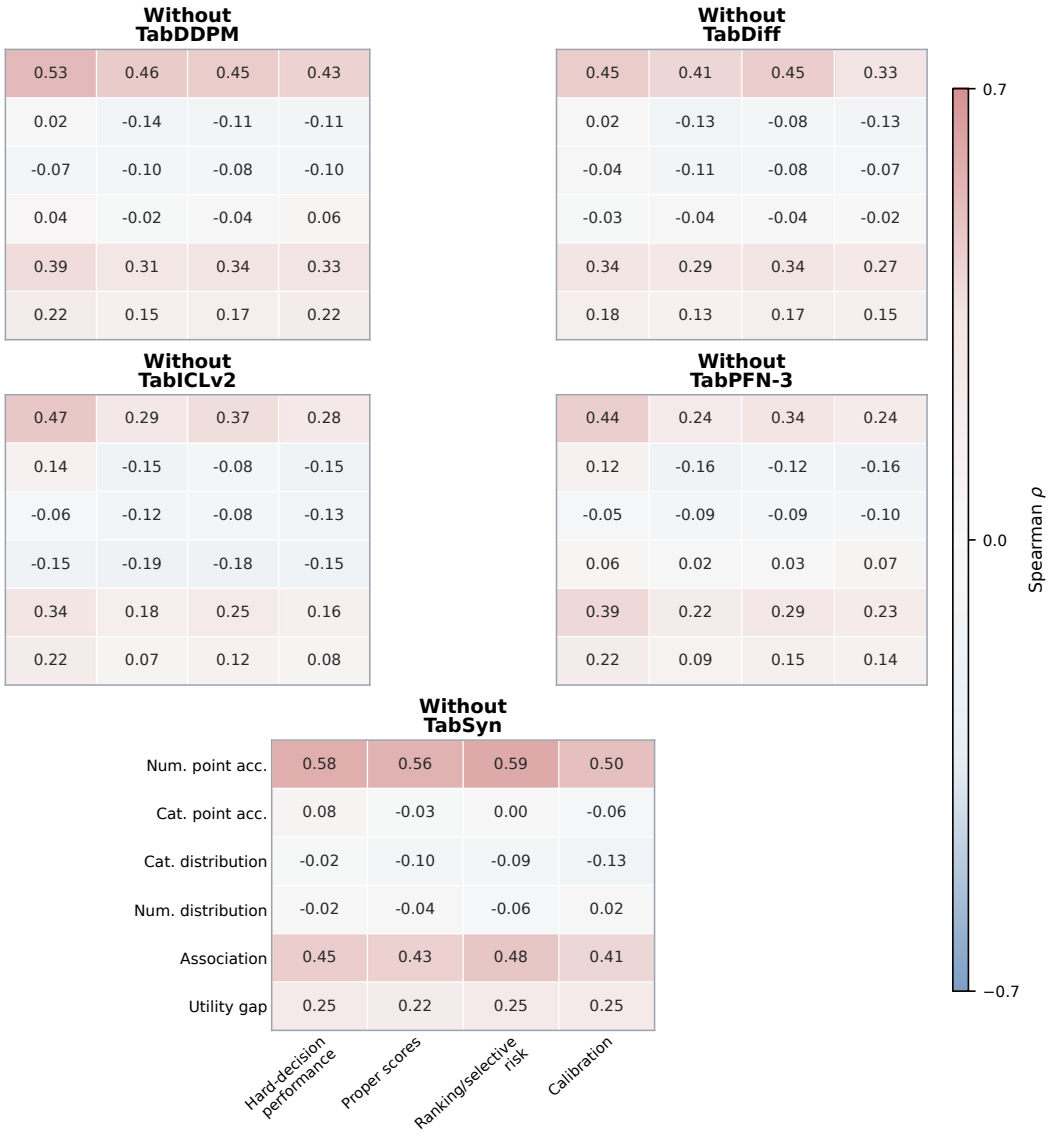


Figure 19: Complete leave-one-method-out results for I × P-cls (part 2 of 2), continuing Figure 18. The encoding and recomputation procedure are unchanged. A panel can match the baseline when the named deletion is a no-op for that axis pair; these deletions are excluded from the sign-agreement denominator.

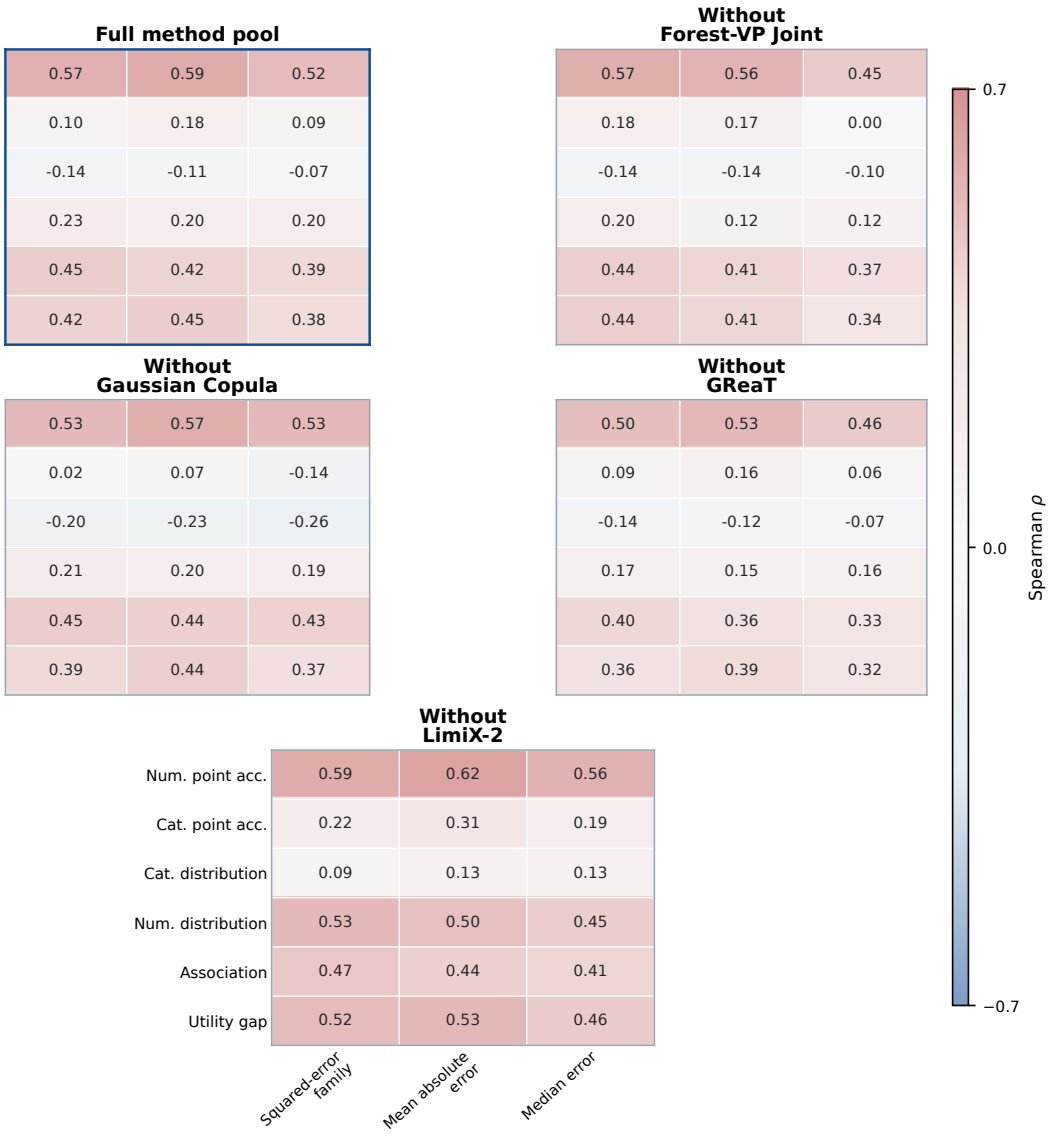


Figure 20: Complete leave-one-method-out results for I × P-reg (part 1 of 2). The "Full method pool" panel is the baseline; every other panel removes the named method. Each cell reports the pooled Spearman correlation. Red and blue denote positive and negative correlations. Support alignment, within-dataset ranks, and capability-axis scores are recomputed for every panel.

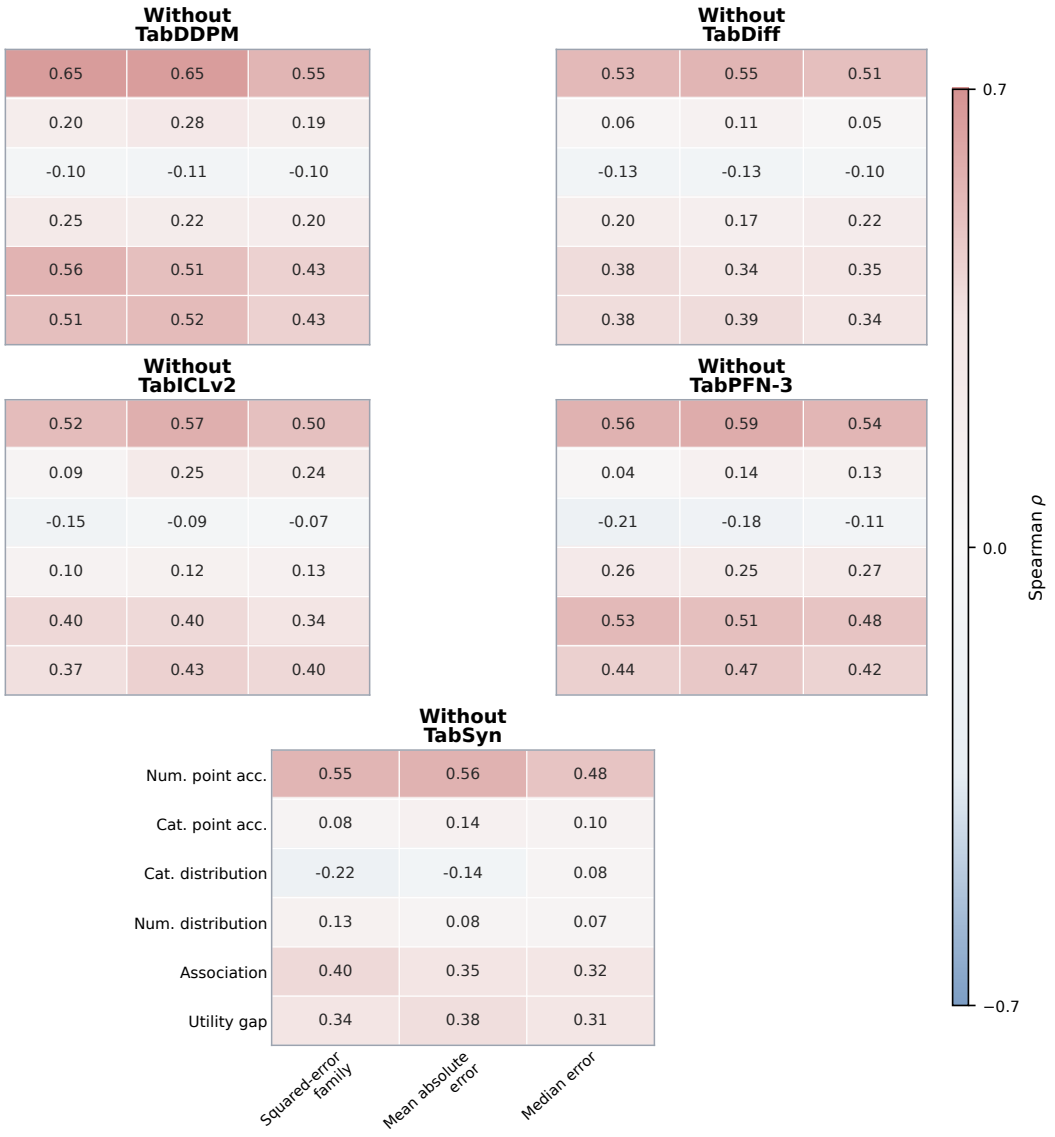


Figure 21: Complete leave-one-method-out results for I × P-reg (part 2 of 2), continuing Figure 20. The encoding and recomputation procedure are unchanged. A panel can match the baseline when the named deletion is a no-op for that axis pair; these deletions are excluded from the sign-agreement denominator.

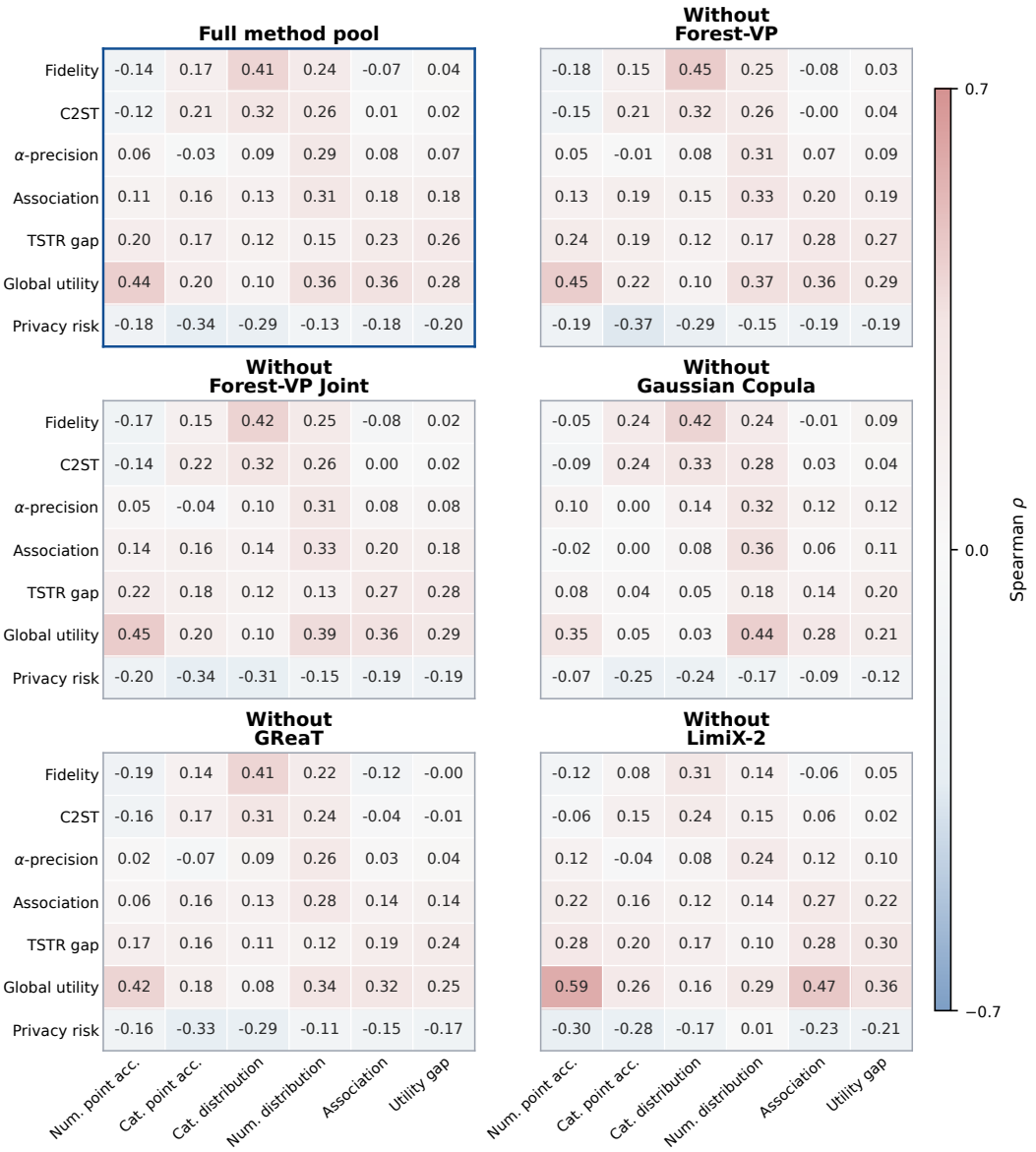


Figure 22: Complete leave-one-method-out results for G × I (part 1 of 2). The "Full method pool" panel is the baseline; every other panel removes the named method. Each cell reports the pooled Spearman correlation. Red and blue denote positive and negative correlations. Support alignment, within-dataset ranks, and capability-axis scores are recomputed for every panel.

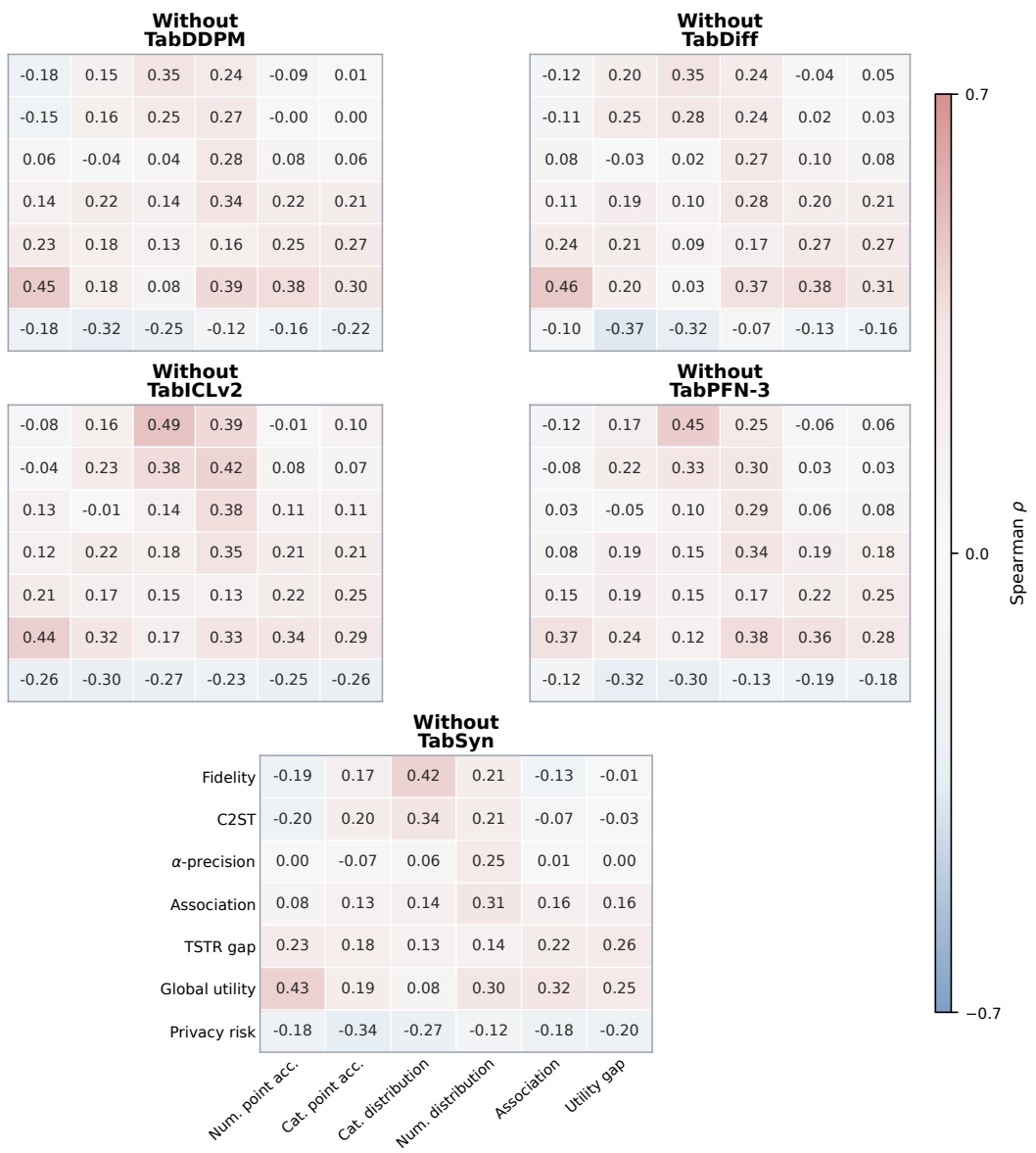


Figure 23: Complete leave-one-method-out results for G × I (part 2 of 2), continuing Figure 22. The encoding and recomputation procedure are unchanged. A panel can match the baseline when the named deletion is a no-op for that axis pair; these deletions are excluded from the sign-agreement denominator.

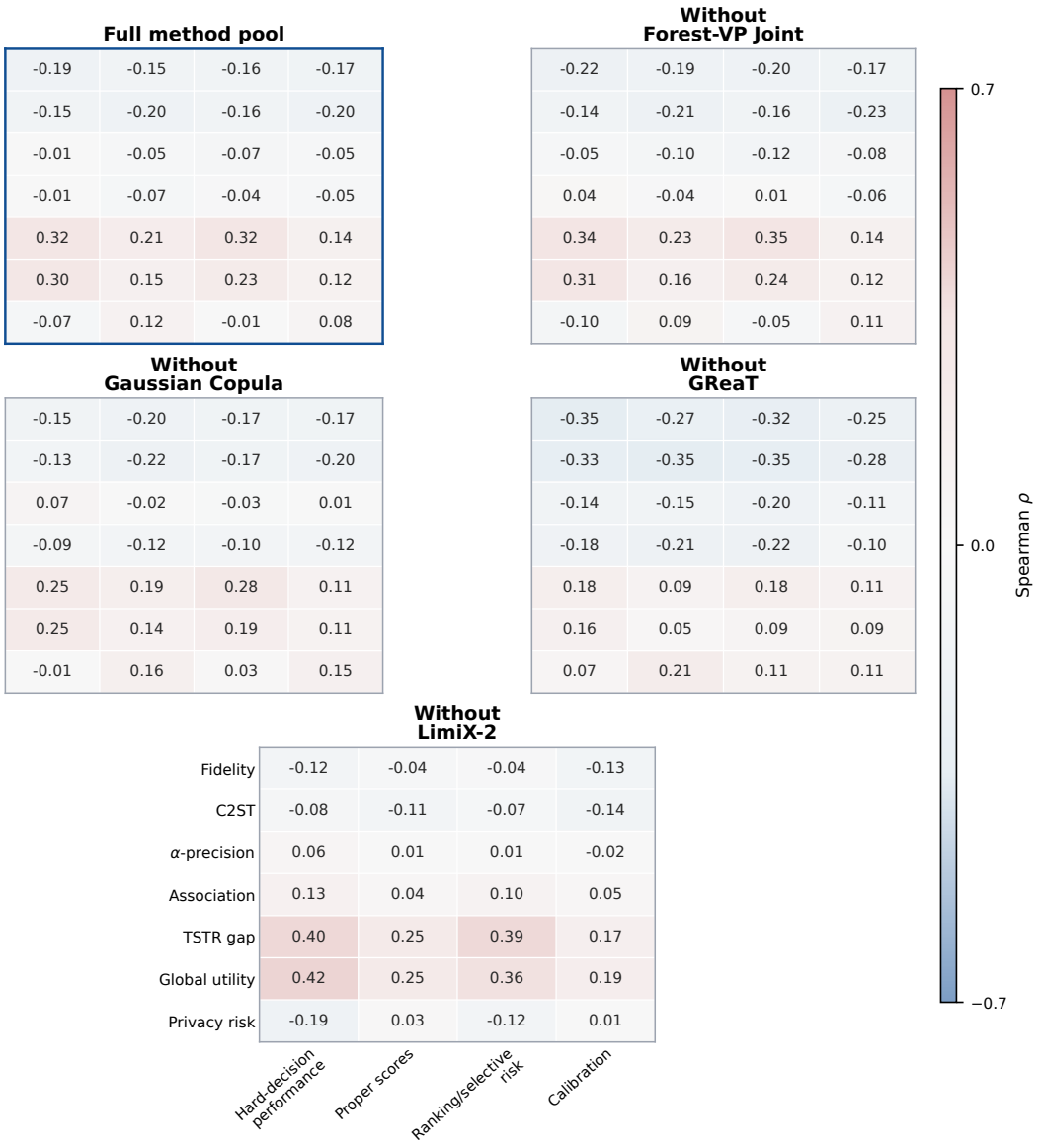


Figure 24: Complete leave-one-method-out results for G × P-cls (part 1 of 2). The "Full method pool" panel is the baseline; every other panel removes the named method. Each cell reports the pooled Spearman correlation. Red and blue denote positive and negative correlations. Support alignment, within-dataset ranks, and capability-axis scores are recomputed for every panel.

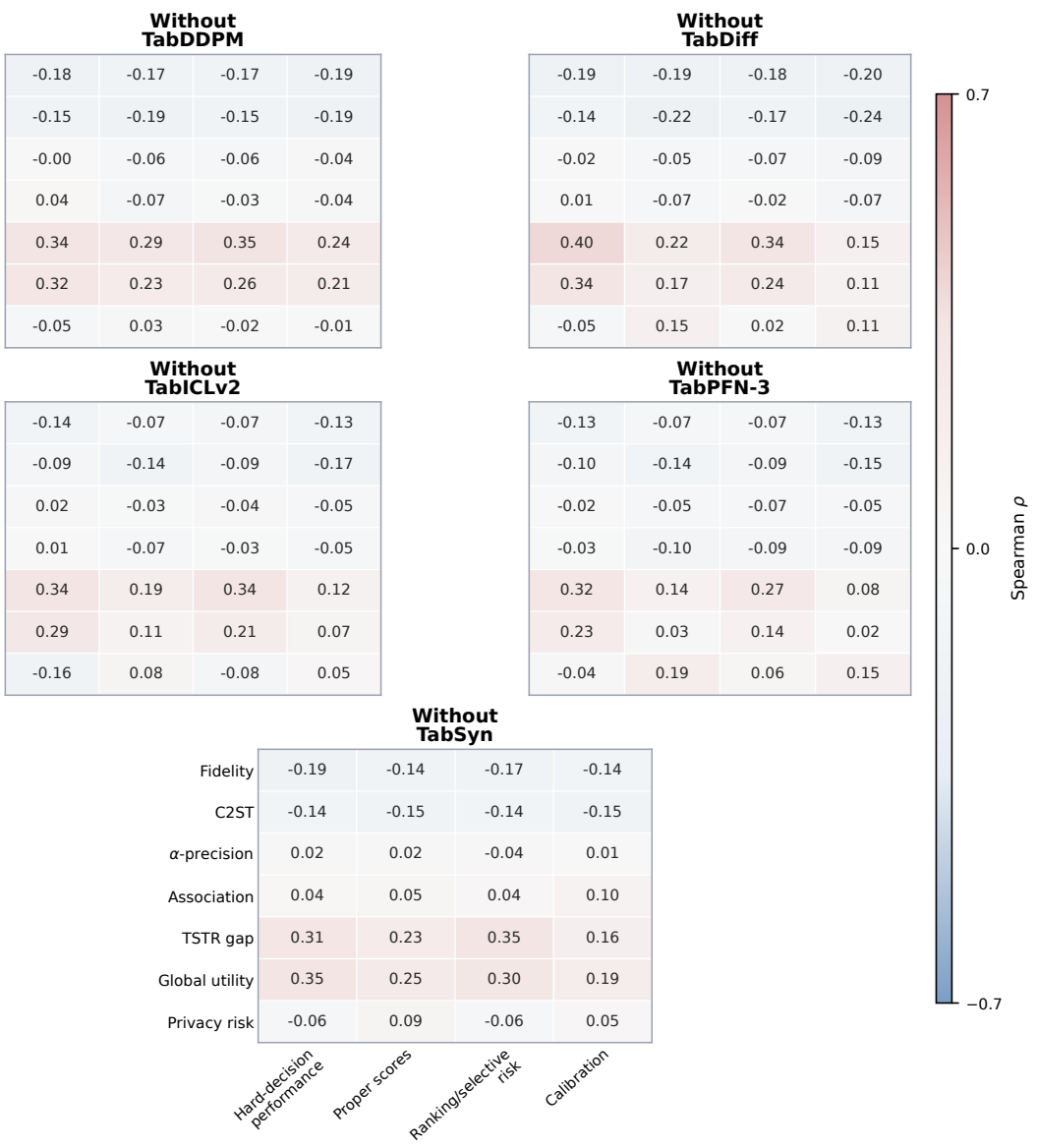


Figure 25: Complete leave-one-method-out results for G × P-cls (part 2 of 2), continuing Figure 24. The encoding and recomputation procedure are unchanged. A panel can match the baseline when the named deletion is a no-op for that axis pair; these deletions are excluded from the sign-agreement denominator.

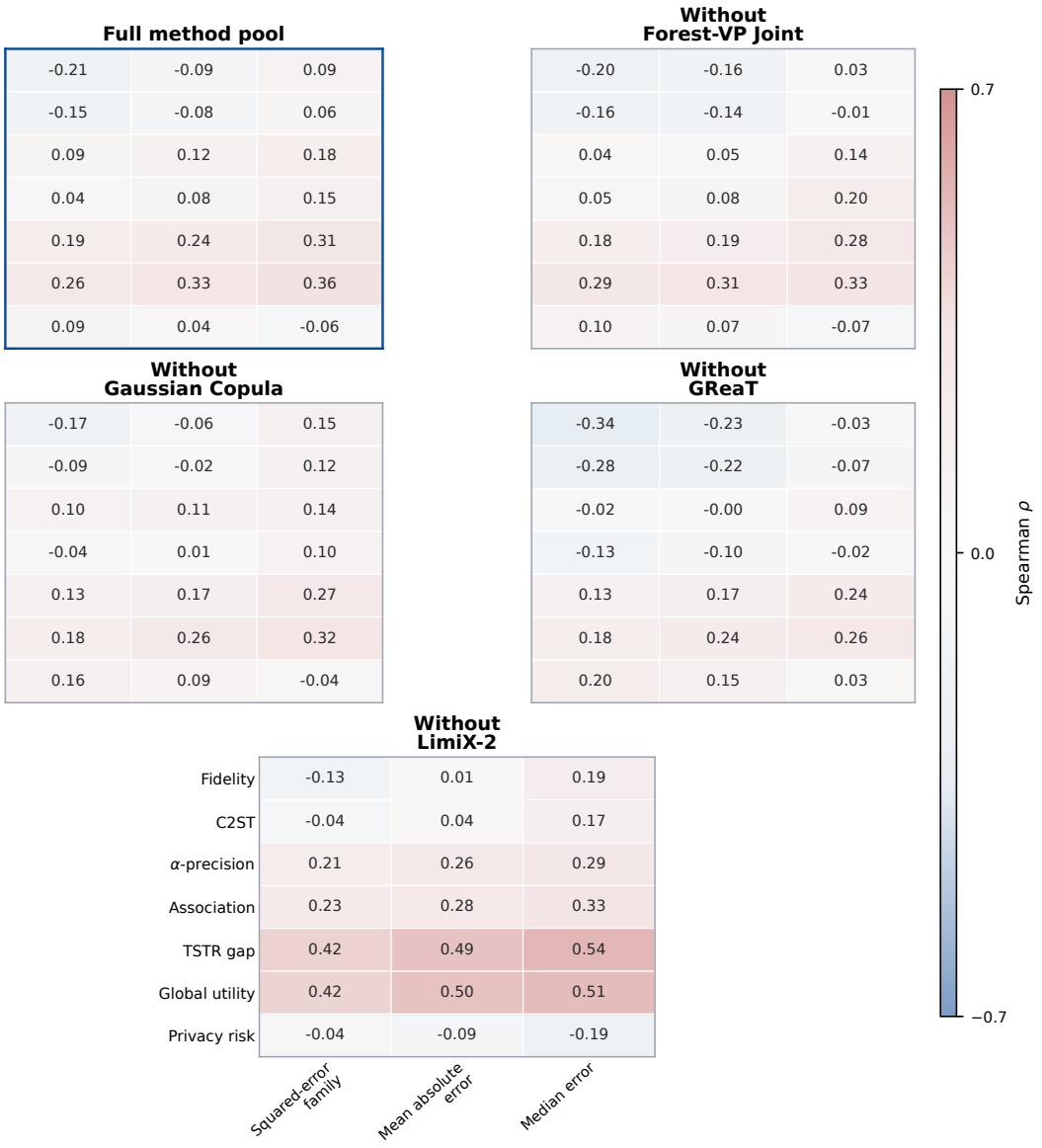


Figure 26: Complete leave-one-method-out results for G × P-reg (part 1 of 2). The "Full method pool" panel is the baseline; every other panel removes the named method. Each cell reports the pooled Spearman correlation. Red and blue denote positive and negative correlations. Support alignment, within-dataset ranks, and capability-axis scores are recomputed for every panel.

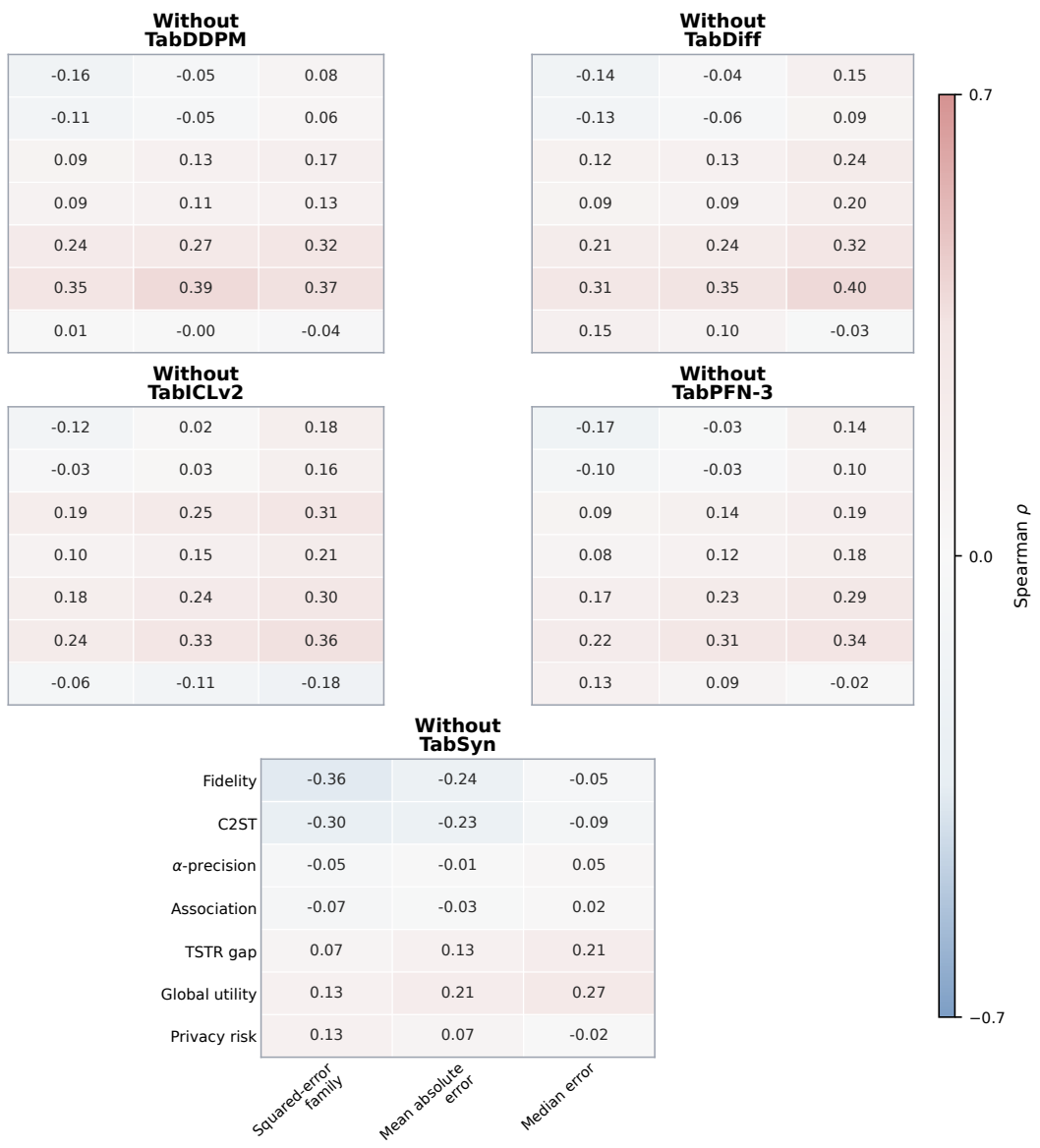


Figure 27: Complete leave-one-method-out results for G × P-reg (part 2 of 2), continuing Figure 26. The encoding and recomputation procedure are unchanged. A panel can match the baseline when the named deletion is a no-op for that axis pair; these deletions are excluded from the sign-agreement denominator.

E CONTROLLED TRANSFER: PROTOCOL AND ADDITIONAL DIAGNOSTICS

This section expands the matched-objective study in Section 5, following the observational alignment analysis in Appendix D. It specifies the intervention and uncertainty protocols, then presents diagnostics of backbone heterogeneity, conditional interactions, and objective-arm Pareto frontiers.

E.1 STRICT MEMBER-COMPLETE CONSTRUCTION

The intervention unit is a fixed backbone–dataset–training-seed block, and the treatments are the seven non-empty objective arms $\mathcal{C} = \{G, I, P, GI, GP, IP, GIP\}$. The classification datasets are JapaneseVowels, cleveland, credit_card_clients_default, and website_phishing; MiamiHousing2016 is the regression dataset. Their OpenML identifiers and fixed outer splits are given in Table 15. Within each atomic evaluation setting, we orient every metric member so that larger is better and assign average-tie ranks across exactly these arms, normalized to $[0, 1]$. Members receive equal weight within a semantic axis, and an axis score is computed only when every declared member is finite for every arm. Imputation is ranked and matched separately within each missingness mechanism and rate; setting-level contrasts are then averaged with equal weight. Structural applicability is defined by the task contract.

The causal population comprises Gaussian Copula, Forest-VP Joint, TabDiff, TabSyn, and TabDDPM. Forest-VP Joint uses native query-masked trees for the seven treatments: unqueried tasks contribute neither gradients nor Hessians. GReaT is not part of the factorial contrasts.

E.2 UNCERTAINTY AND ROBUSTNESS

Confidence intervals use 2,000 crossed-bootstrap draws (Efron, 1979; Owen & Eckles, 2012) that resample backbones and datasets while resampling seeds within the selected blocks; all objective arms remain paired. Two-sided Wilcoxon diagnostics (Wilcoxon, 1945) are Holm-adjusted (Holm, 1979) within each declared analysis family. An effect receives the robust marker only when its bootstrap interval excludes zero, its adjusted p-value is below $.05$, its magnitude is at least $.05$ rank units, the required backbone evidence is complete on multiple datasets, and backbone-specific means share a non-zero sign. Dataset-sign and leave-one-backbone summaries are exported as diagnostics but do not enter this gate.

Generation self-inclusion passes this gate on fidelity, C2ST, α-precision, association, TSTR, and global utility; imputation self-inclusion passes on association, categorical distribution, numerical point accuracy, and downstream utility. These manipulation checks show that the intervention changes its intended recipients. Under the primary within-backbone seven-arm rank contract, no retained-axis off-diagonal source–recipient contrast passes the same gate.

E.3 UTILITY SCOPE AND BACKBONE HETEROGENEITY

For P→G, the retained TSTR estimate is 0.111 $[-0.012, 0.249]$, while the global-utility estimate is -0.007 $[-0.047, 0.042]$. Scope-matched target, all-column, and non-target diagnostics form a separate multiplicity family. Their direct paired localization contrasts remain unresolved: target-minus-all is 0.086 $[-0.039, 0.244]$ and target-minus-non-target is 0.087 $[-0.057, 0.265]$. Figure 28 also shows the backbone variation obscured by the pooled estimates: Forest-VP Joint's TSTR effect has the opposite sign from those of several other backbones.

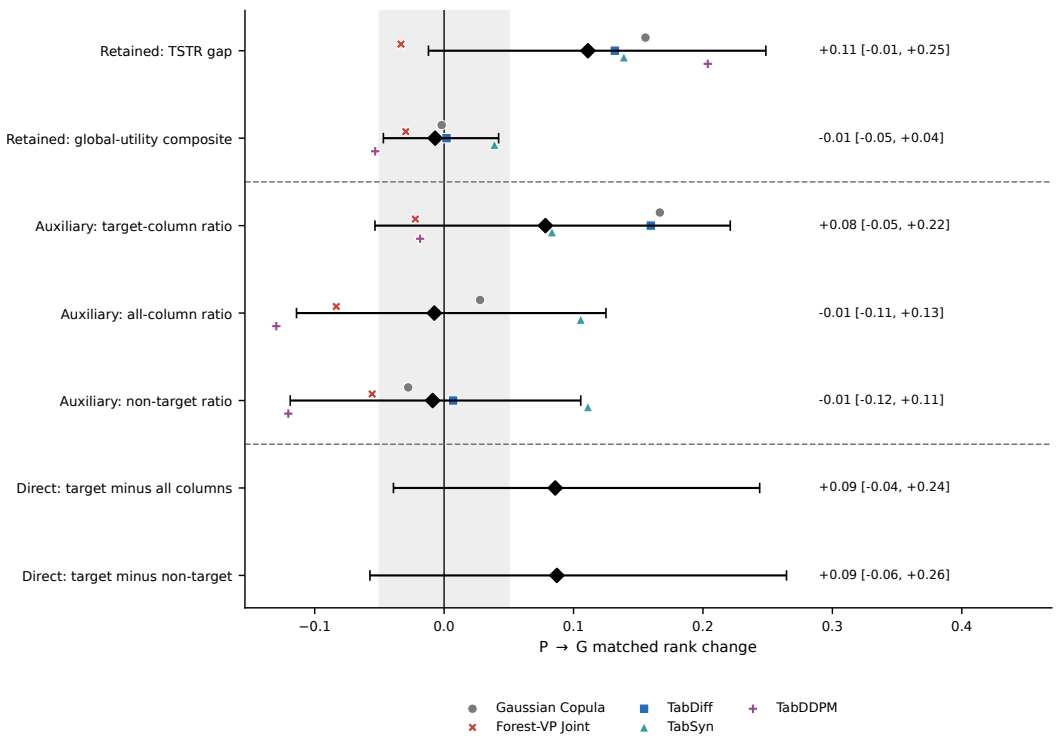


Figure 28: **Scope of prediction-to-generation utility responses.** Diamonds and intervals denote pooled matched effects and 95% bootstrap intervals; colored markers denote backbone means. The upper rows are retained generation axes, and the lower rows are auxiliary diagnostics computed by one scope-matched per-column evaluator. Localization is evaluated by the paired contrasts reported in the text, not by comparing whether separate intervals cross zero.

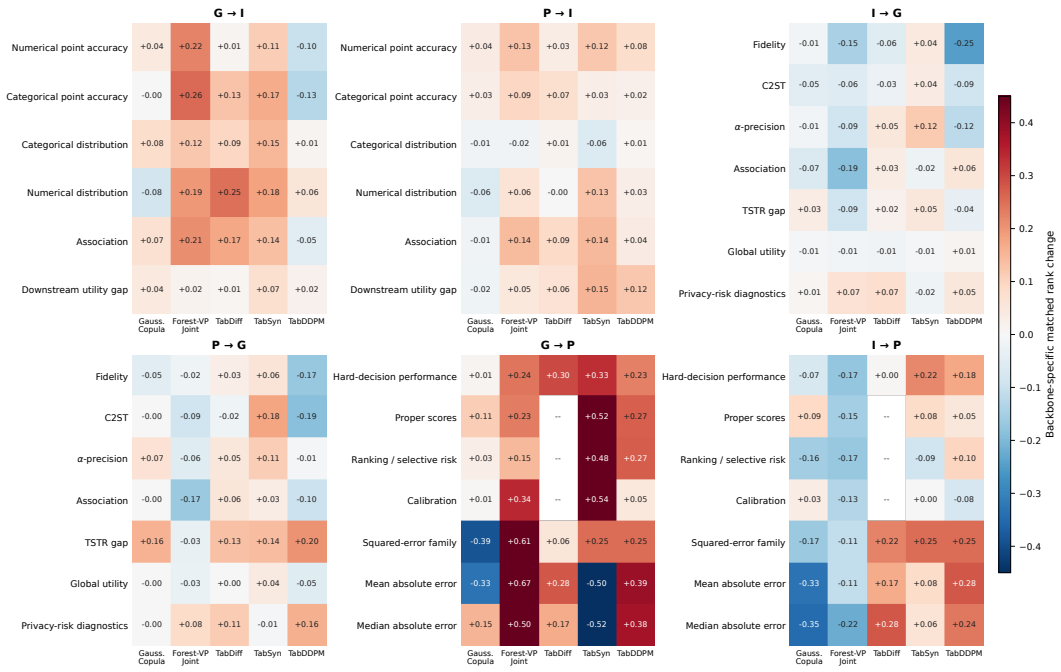


Figure 29: **Backbone heterogeneity in recipient-conditioned transfer.** Each cell is a backbone-specific matched rank change for one source–recipient axis. These panels diagnose sign and magnitude variation behind the pooled estimates; they do not define an additional family of primary causal claims.

Imputation mechanism and rate remain atomic throughout ranking and matching. Figure 30 shows that positive pooled tendencies are not confined to a single missingness regime, but none passes the full robustness gate after accounting for backbone variation and multiplicity.

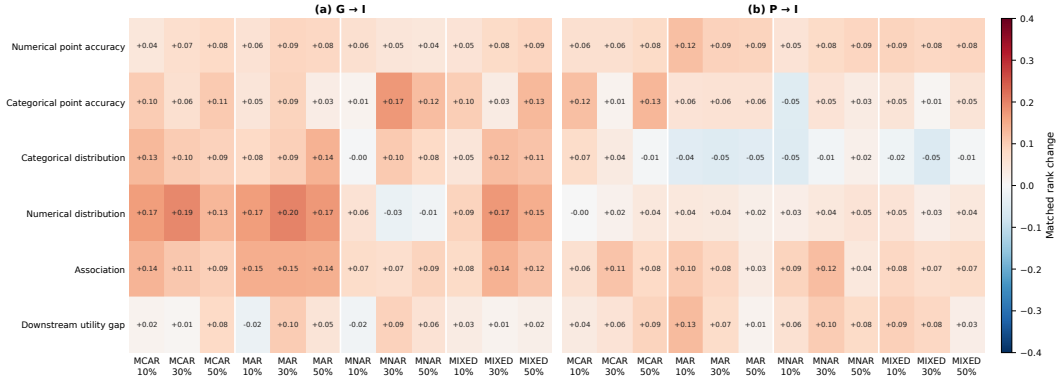


Figure 30: **Transfer into imputation across missingness settings.** The two panels add G or P to an imputation recipient. Every mechanism–rate setting is ranked and matched separately before equally weighted averaging; blank cells, if present, denote unavailable outcomes rather than zero.

E.4 Conditional interactions

Figure 31 gives all anchor-conditional differences-in-differences. Five P-anchored G–I interactions pass the gate: generation C2ST (-0.320), fidelity (-0.356), and global utility (-0.119), together with imputation association (-0.449) and numerical point accuracy (-0.526). No retained GIP-versus-singleton comparison establishes a robust universal advantage or tax. These estimates describe sub-additivity for the implemented anchored objective mixture.

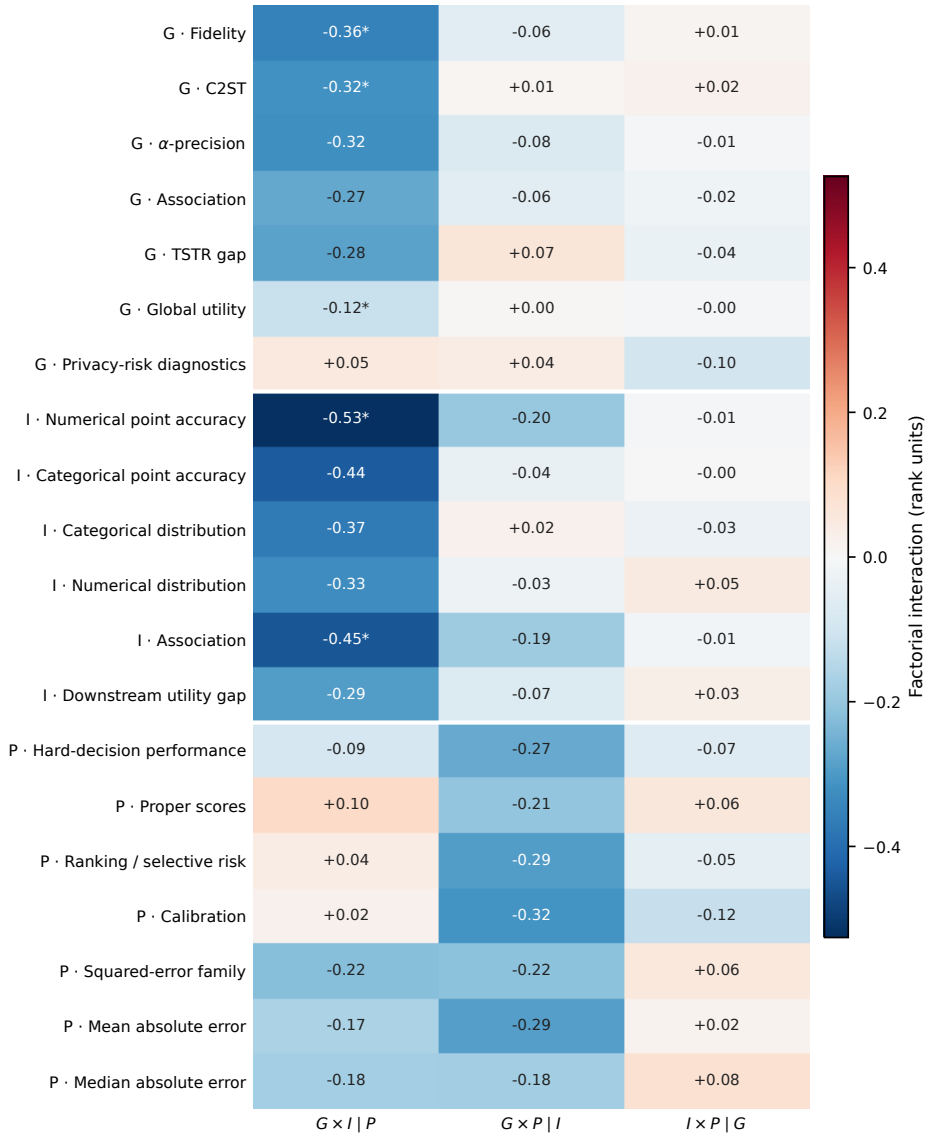


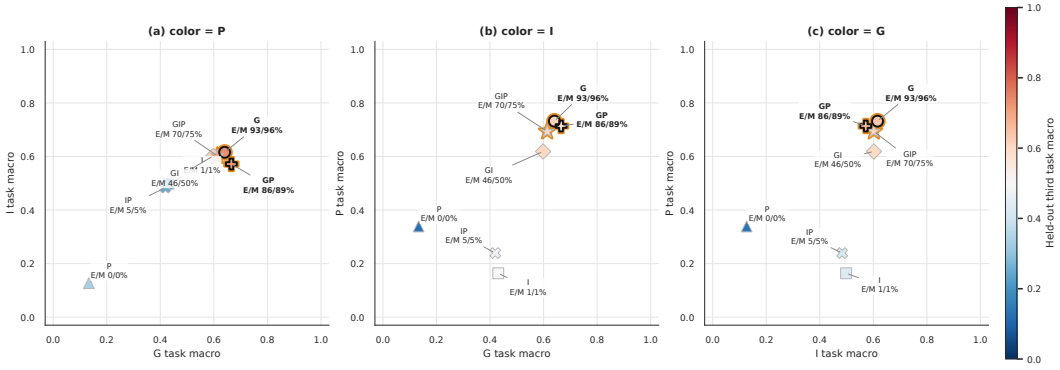
Figure 31: **Anchor-conditional objective interactions.** Cells are differences-in-differences on the normalized seven-arm rank scale. Positive values indicate complementarity and negative values sub-additivity; * marks passage of the complete robustness gate. The first column is $G \times I \mid P$, the second $G \times P \mid I$, and the third $I \times P \mid G$. Blank cells, if present, are unavailable rather than zero.

E.5 Strict objective-arm Pareto construction

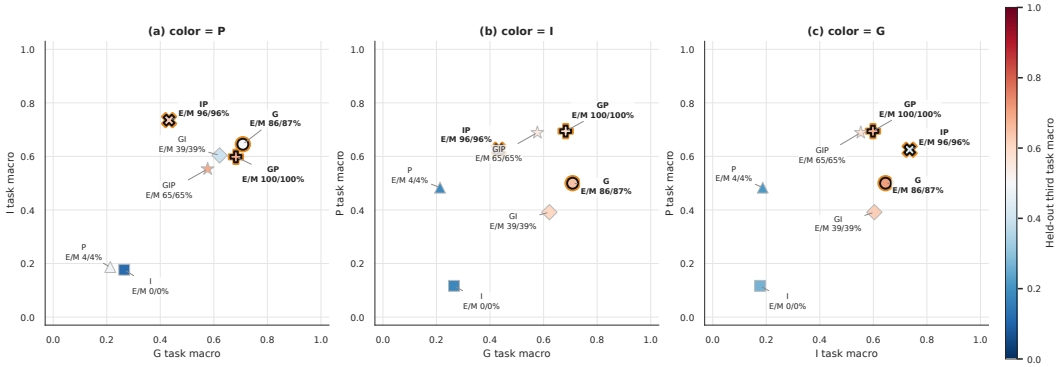
The Pareto candidates are the seven objective arms. Within each matched block, member ranks are averaged within semantic axes, and the applicable axis scores are then averaged with equal weight to form G, I, and P task macro-scores. Imputation first averages its setting-specific axis scores with equal weight. Classification and regression are analyzed separately, and every coordinate in a given analysis is computed on the same rectangular member-complete subset. Exact dominance requires no loss on any task and a strict gain on at least one; material dominance additionally requires a gain of at least $\epsilon = 0.05$ without a loss on the other coordinates.

The classification exact frontier is $\{G, GP\}$ and its material frontier is $\{G, GP, GIP\}$; the regression exact and material frontiers are both $\{G, GP, IP\}$. Figure 32 shows all pairwise projections, but frontier membership is computed once in three dimensions. Bootstrap membership frequencies

measure the stability of these discrete sets under paired resampling; they are neither posterior probabilities of optimality nor significance levels.



(a) Full-axis classification.



(b) Regression.

Figure 32: **Three-task objective-arm Pareto geometry.** Each row gives the three pairwise projections of one frontier computed jointly in (G, I, P). Black outlines mark the exact frontier, orange halos the material frontier, and E/M labels the bootstrap exact/material membership frequencies.

Under the hard-decision-only classification contract and its corresponding strict subset, both frontiers expand to $\{G, GI, GP, GIP\}$. A second sensitivity ranks all backbone–arm configurations in one global denominator and changes one off-diagonal label (G→P hard-decision). Because that denominator combines between-backbone differences with objective-arm variation, it does not replace the primary within-backbone matched-arm estimand. Treating privacy risk as a fourth coordinate, checking exact-copy rate as a descriptive guardrail, and leave-one-backbone recomputations are also retained in the released analysis bundle. Across the leave-one-backbone analyses, GP is the most stable observed compromise; this does not make it universally optimal.

F FIXED-METHOD GENERALITY: PROTOCOL AND ADDITIONAL DIAGNOSTICS

This section expands the fixed-identity deployment study in Section 6. Following the main-text order, it presents strict estimands and specialist references, cross-task profile geometry, and portfolio and operational diagnostics.

F.1 STRICT ESTIMANDS AND FIXED-SPECIALIST BASELINES

The method analysis uses the 20 semantic axes and 42 metric members specified in Table 3. Replicates are aggregated first; each member is oriented and converted to percentile ranks with average ties within its protocol instance; and members receive equal weight under the member-complete construction. Target-observed and feature-only imputation are never placed in one denominator. The primary imputation estimand remains separate for every mechanism–rate setting; the cross-task display coordinate is their equally weighted mean, without reranking.

A fixed specialist keeps one identity across every dataset and axis in its profile. Exact dominance is evaluated jointly in the full profile, and practical dominance permits an $\epsilon = 0.05$ tolerance on every coordinate. Uncertainty is estimated using 2,000 bootstrap draws of whole datasets. Pairwise tests remove numerical ties at 10^{-12}, use a two-sided Wilcoxon signed-rank test, and apply Holm correction within each declared comparison family. Frontier frequency is a stability diagnostic, not a probability that a method is optimal. The historical per-cell oracle is retained only as an unattainable supplement.

The fixed-specialist pool includes ARF (Watson et al., 2023), MissForest (Stekhoven & Bühlmann, 2012), DiffPuter (Zhang et al., 2025a), KNN imputation (Troyanskaya et al., 2001), MICE (van Buuren & Groothuis-Oudshoorn, 2011), ReMasker (Du et al., 2024), CatBoost (Prokhorenkova et al., 2018), TabM (Gorishniy et al., 2025), LightGBM (Ke et al., 2017), and Random Forest (Breiman, 2001).

Table 13: **Fixed specialist reference frontiers.** For imputation, $x/12$ is the number of separately evaluated mechanism–rate settings in which the method belongs to the indicated frontier; it is not a dataset-support count. The provisional generation privacy composite is excluded.

Task profile	Exact frontier	$\epsilon = 0.05$ frontier
Generation, six specified axes	ARF	ARF
Imputation, numerical targets	MissForest $12/12$; DiffPuter $12/12$; KNN Imputer $4/12$; MICE $1/12$	MissForest $12/12$; DiffPuter $11/12$; KNN Imputer $2/12$
Imputation, categorical targets	MissForest $11/12$; MICE $10/12$; DiffPuter $6/12$; KNN Imputer $2/12$; ReMasker $2/12$	MissForest $11/12$; MICE $7/12$; DiffPuter $3/12$
Prediction, classification	CatBoost; TabM	CatBoost; TabM
Prediction, regression	LightGBM; LightGBM Quantile; Random Forest	LightGBM Quantile; Random Forest

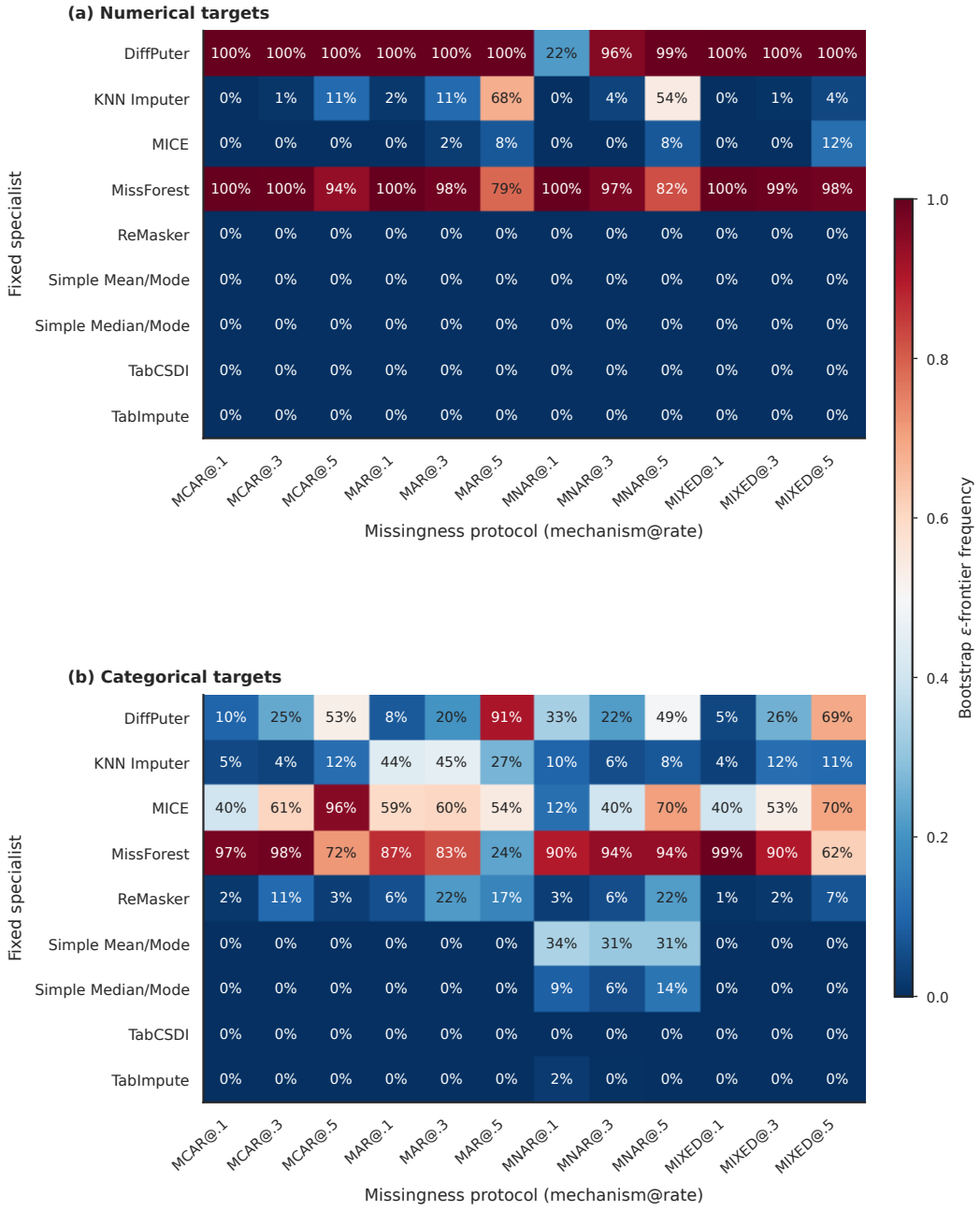


Figure 33: **Strict imputation specialist-frontier stability.** The top panel evaluates numerical targets and the bottom panel categorical targets. Columns are the twelve target-observed mechanism–rate settings; cells give the frequency of membership in the $\epsilon = 0.05$ fixed-specialist frontier under whole-dataset bootstrap resampling. Settings are not pooled when computing the frontier.

MissForest is practically stable across all numerical settings, while DiffPuter is nearly as persistent; their advantage is therefore not an artifact of one missingness regime. Categorical imputation is less uniform: MissForest remains the most persistent reference, but MICE and DiffPuter enter the frontier in setting-dependent regions. Generation results involving the normalized median Gower DCR use the provisional target-one transformation $-|x - 1|$ and are excluded from confirmatory privacy claims.

F.2 CROSS-TASK PROFILE GEOMETRY AND CONTROLS

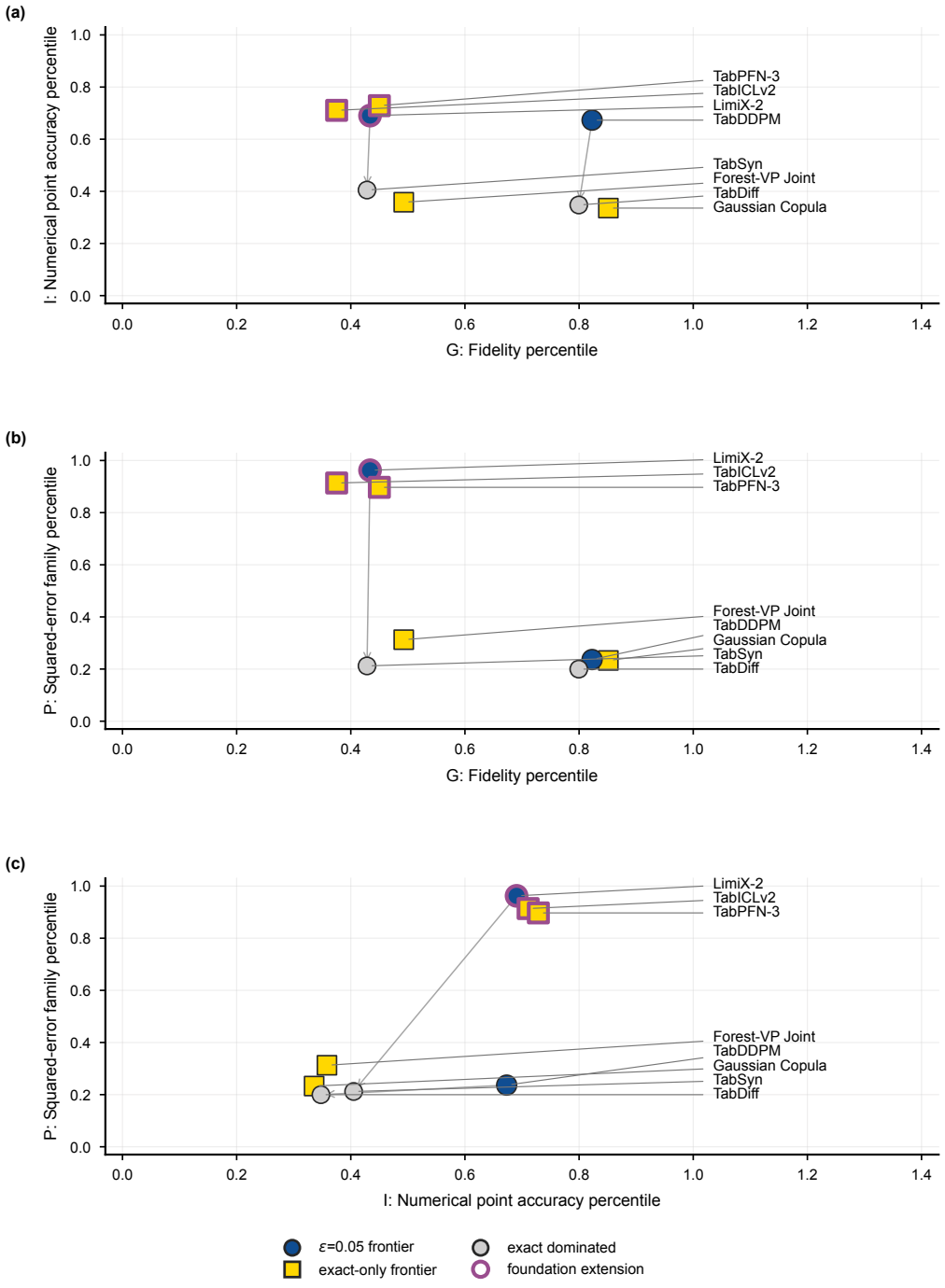


Figure 34: **Pairwise projections of the numerical-fidelity profile.** From top to bottom, the panels show the G–I, G–P, and I–P projections; dominance is always computed jointly in all three coordinates. Blue circles are on the practical frontier, yellow squares are exact-only frontier points, gray circles are exactly dominated, and arrows show one representative dominance certificate.

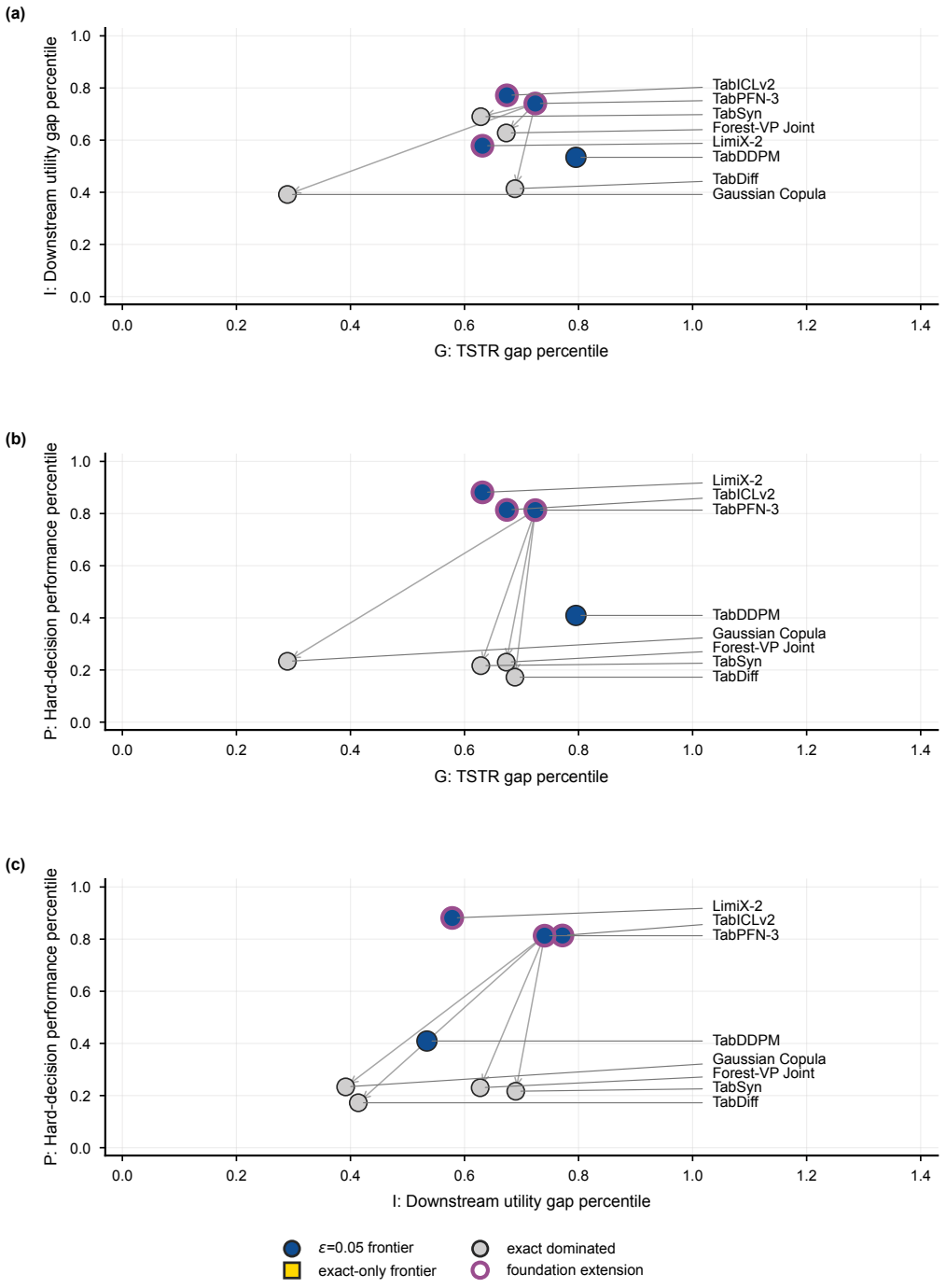


Figure 35: **Pairwise projections of the downstream-utility profile.** From top to bottom, the panels show the G–I, G–P, and I–P projections; symbols follow Figure 34, and dominance is computed in the full three-coordinate profile.

The numerical-fidelity exact frontier contains Forest-VP Joint, Gaussian Copula, LimiX-2, TabDDPM, TabICLv2, and TabPFN-3, but only LimiX-2 and TabDDPM remain after the $\epsilon = 0.05$ practical margin. The downstream-utility exact and practical frontiers both contain LimiX-2, TabDDPM, TabICLv2, and TabPFN-3. These sets are computed in three dimensions; the projections above are explanatory views rather than separate frontier calculations.

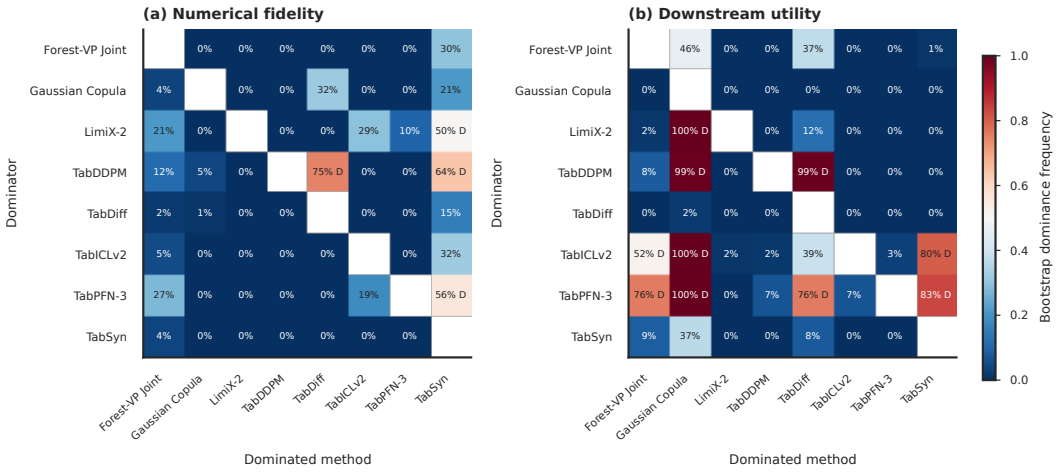


Figure 36: **Stability of named cross-task dominance relations.** Panel (a) is numerical fidelity and panel (b) downstream utility. Rows are candidate dominators, columns are dominated methods, and cells give the whole-dataset bootstrap frequency of exact three-dimensional dominance. "D" marks dominance in the frozen point estimate.

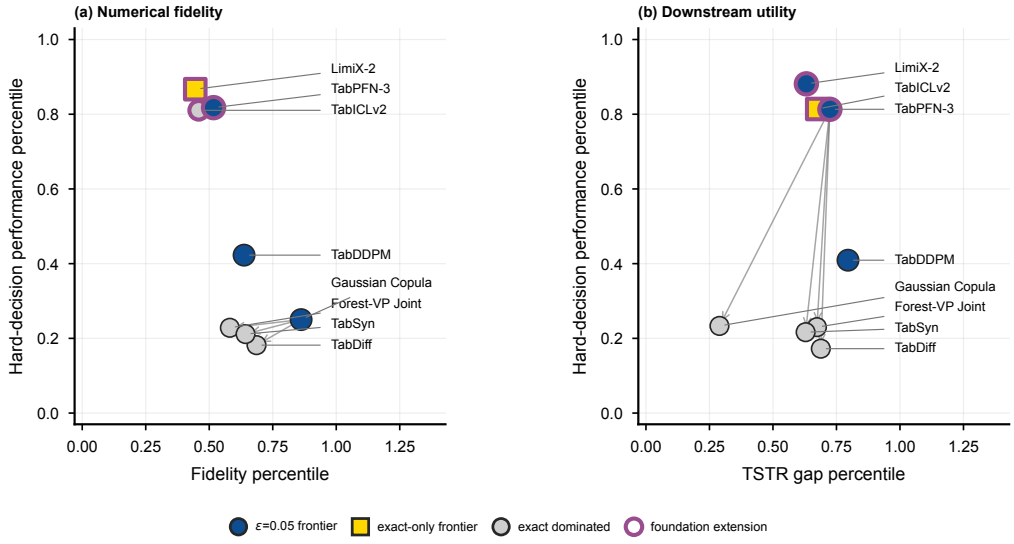


Figure 37: **Generation–prediction controls without an imputation coordinate.** Panel (a) compares generation fidelity with classification hard decisions; panel (b) compares TSTR utility with the same prediction axis. These two-dimensional controls avoid the supplementary imputation summary.

F.3 PORTFOLIO AND OPERATIONAL DIAGNOSTICS

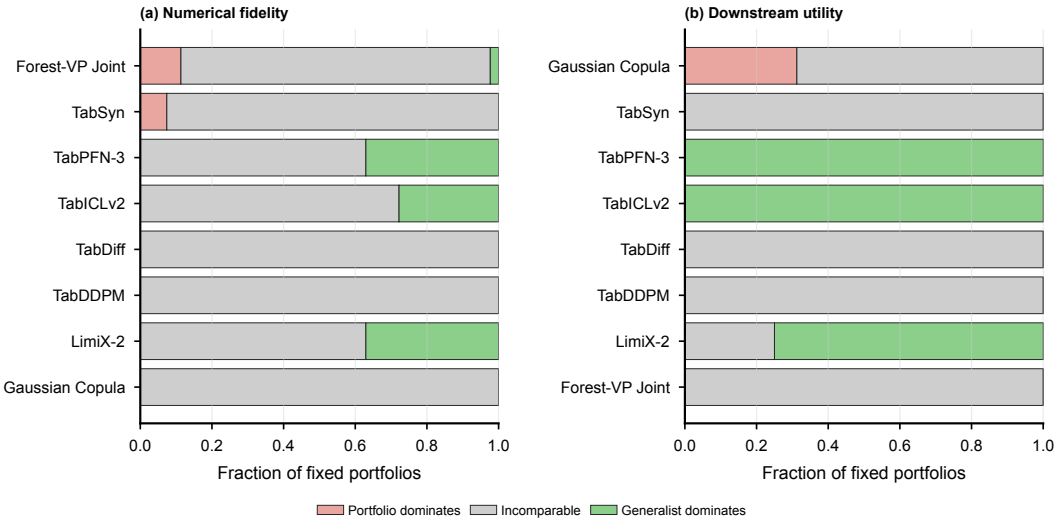


Figure 38: **In-sample relation between fixed generalists and fixed specialist portfolios.** Panel (a) is numerical fidelity and panel (b) downstream utility. Bars partition prespecified specialist triples into those that dominate the generalist, are dominated by it, or remain incomparable in the joint three-task geometry. This maps available trade-offs; it is not a held-out selection result.

For LODO, the most frequently selected numerical portfolio combines ARF, MissForest, and LightGBM; the utility selection combines ARF, DiffPuter, and Random Forest in nearly every fold. Candidate methods are fixed before LODO. Bootstrap intervals summarize the held-out gaps conditional on the selected portfolios, with training datasets shared across folds.

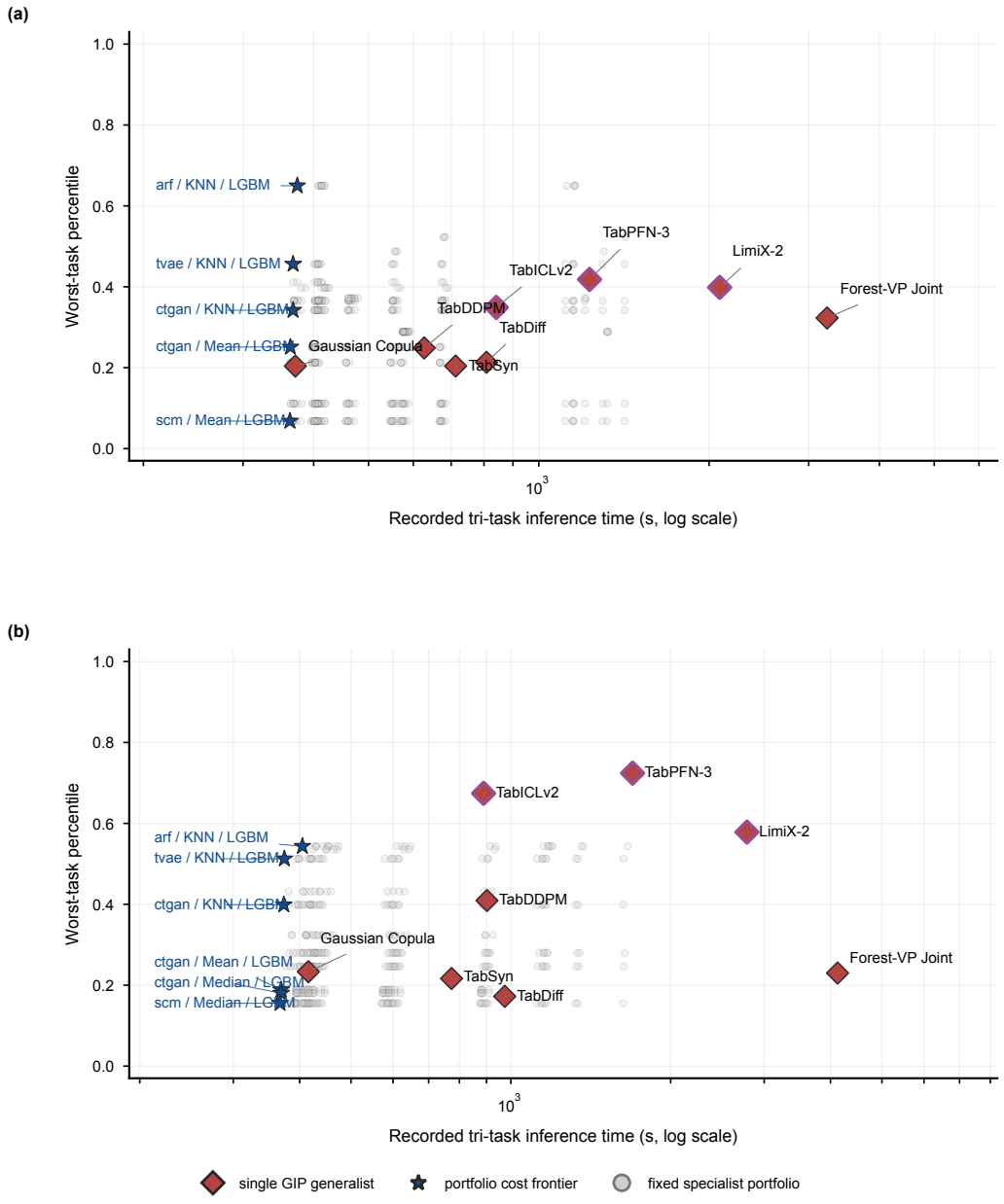


Figure 39: **Diagnostic quality–cost view.** The top panel is numerical fidelity and the bottom panel downstream utility. The vertical coordinate is worst-task percentile and the horizontal coordinate is tri-task inference time under a balanced workload. Stars mark the fixed-portfolio cost frontier and diamonds mark generalists. Costs refer to the displayed inference workload and serve as an operational diagnostic.

F.4 Interpretation rules

We use "associated" only for observational rank correlations, "transfers" only for matched objective contrasts, "dominates" only for a declared Pareto profile, and "LODO robustness" only for held-out dataset comparison under a fixed selection protocol. Criterion-overlap rows are retained as controls but do not count as independent cross-task evidence.

G LIMITATIONS AND FUTURE WORK

Scope of cross-task relations. The evidence supports neither "all three tasks are the same" nor "all three are unrelated." Each relationship is specific to a source, recipient, capability axis, contract, and backbone population. This graph is a more useful target for model design: it identifies where shared training produces broad structure, where apparent transfer is backbone-dependent, and where losses interfere.

Semantic and statistical criteria for metric reduction. Absolute correlation can place opposing ends of a trade-off in one cluster. Reliability correction can also merge metrics whose observed correlations are attenuated by noise. We therefore publish the raw catalog, redundancy map, representatives, and retained trade-off pairs. If a future aggregate index is used, it should express a user-declared preference over these axes rather than a hidden average chosen by the benchmark authors.

Current evidence boundaries. Cross-task rank correlations remain descriptive and may reflect shared architectures or metric construction rather than objective transfer. Member-complete alignment fixes the metric composition of each capability axis and aligns methods satisfying both axis contracts. Protected-group evaluation remains outside the current protocol. Equal update budgets do not guarantee equal information per objective. In the method landscape, strict imputation results are setting-specific, whereas the GIP and portfolio views use a supplementary summary across those settings.

Future extensions. Future extensions include a broader factorial backbone set, protected-group metadata for fairness, and measured end-to-end cost under a common hardware contract. These additions would strengthen specific edges of the evidence graph without changing the paper's organizing framework.

H IMPLEMENTATION DETAILS

This appendix specifies the baseline settings, randomization conventions, controlled-transfer configurations, and computational environment. The supplementary source package contains separate code trees for the observational benchmark (`code/tabtoolkit_v1`) and the controlled study (`code/tabtoolkit_v1_causaltransfer`), including their method configurations, evaluation code, and analysis scripts.

H.1 BASELINE CONFIGURATIONS AND EVALUATION PROTOCOL

Configuration policy. Each baseline starts from a method-specific TOML configuration, with experiment-specific overrides applied before fitting. The benchmark runners instantiate these settings directly rather than running a shared external hyperparameter search. Table 14 summarizes the principal default settings for architecture, optimization, and inference; the accompanying configuration files specify every parameter. HyperImpute retains its native model selection for each column and iteration. Its classifier candidates are random forest, logistic regression, XGBoost, CatBoost, and neural networks; its regressor candidates are random forest, linear regression, XGBoost, CatBoost, and neural networks. It uses ten inner iterations, lazy selection, and an optimization threshold of 1,000. It fits each incomplete evaluation matrix through its transductive `fit_transform` interface.

Training and validation. The data store supplies the fixed train–test split described in Appendix A. TabDDPM, TabDiff, and TabSyn use a deterministic 10% validation subset of the training rows, selected with seed 0. Their training-data view redirects any upstream training-time "test" access to this validation subset. XGBoost, LightGBM, CatBoost, NGBoost, LightGBM-Quantile, FT-Transformer, TabM, and CARD's guidance network use a 20% internal validation split determined by the training seed. FT-Transformer and TabM use AdamW and retain the state with the lowest validation loss; their patience is 16 epochs. MLP uses scikit-learn's internal early-stopping split. Ridge, logistic regression, and random forest fit their training rows without an additional benchmark validation split.

Randomization and evaluation settings. The seed set is $\{0, 1, 2\}$. Generation uses the same seed for training and sampling. Imputation uses paired seed tuples $(s_{\text{train}}, s_{\text{selection}}, s_{\text{mask}}, s_{\text{impute}}) = (s, s, s, s)$ for $s \in \{0, 1, 2\}$, across MCAR, MAR, MNAR, and mixed MCAR/MAR/MNAR at rates $\{0.1, 0.3, 0.5\}$. Prediction is evaluated separately for each training seed. Training seeds and conditional-sampling draws are distinct randomization levels.

Table 14: Principal default baseline settings. Learning rates are denoted by LR, weight decay by WD, and batch size by B. Budgets are expressed in the units used by each implementation. Full per-method TOML configurations accompany the benchmark.

Method	Configuration
Methods with generation or conditional-query interfaces	
TabDDPM	MLP widths $[1024, 2048, 2048, 1024]$; time embedding 128; 1,000 diffusion steps; cosine schedule; 100,000 updates, LR 10^{-3}, WD 5×10^{-4}, $B = 4096$; quantile normalization.
TabDiff	Two layers, token width 4, one head, factor 32, time width 1,024; 50 sampling steps; 8,000 training epochs, LR 10^{-3}, EMA 0.997, $B = 4096$; main and target-only prediction-guidance models.
TabSyn	Two-layer, width-4 VAE tokens; latent diffusion width 1,024; VAE 4,000 epochs and diffusion 10,001 epochs, LR 10^{-3}, $B = 4096$, patience 500; 50 sampling steps, 20 imputation resampling rounds.
Forest-VP / Joint	50 time points, 100 noisy duplicates, 100 trees, depth 7, shrinkage 0.3; histogram trees; RePaint $r = 5$, $j = 0.1$; inference row batches 256 (I) and 1,024 (G).
Gaussian copula	At most 1,000 marginal quantiles; 100 conditional samples for regression and 200 class-conditional samples.
GReaT	Training iteration parameter 5; at most 3,000 training rows; completion temperature 0.7, batch 100, maximum length 1,000, and at most 15 retries.
CTGAN	Generator/discriminator: two hidden layers of width 500; 100 training iterations, one discriminator step, LR 10^{-3}, WD 10^{-3}.

Table 14 (continued)

Method	Configuration
TVAE	Encoder/decoder: three hidden layers of width 500; embedding width 500; 100 training iterations, LR 10^{-3}, WD 10^{-5}.
NFlow	One hidden layer of width 100; rational-quadratic autoregressive transform with permutation; dropout 0.1; 1,000 iterations, LR 10^{-3}; at most 2,000 training rows.
ARF	30 trees, minimum node size 5, at most ten adversarial iterations, early stopping enabled.
Bayesian network	Tree structure search with K2 score; structure-learning iteration parameter 1,000; at most 10,000 training rows; 100-bin numerical discretization.
SCM generator	DirectLiNGAM causal-discovery backend.
NRGBoost	200 trees, shrinkage 0.15, at most 256 leaves; 80,000 model samples, 16 chains, burn-in 100.
TabEBM	200 SGLD steps, step size 0.1, noise standard deviation 0.01; at most 10,000 context rows; chunked one-hot inference.
SMOTE / SMOTE-NC / SMOTER	Five neighbors; task-appropriate numerical, mixed-type, or regression interpolation.
Independent marginals / training-row bootstrap	Empirical per-column sampling / empirical row sampling with replacement.
Imputation specialists	
Mean/mode; median/mode; kNN	Numerical mean or median with categorical mode; kNN uses five neighbors.
MICE / MissForest	Ten chained iterations / at most ten forest iterations with 100 trees and tolerance 10^{-3}.
HyperImpute	Five classifier and five regressor candidates; ten inner iterations; model selection by column and iteration; lazy selection threshold 1,000.
ReMasker	Encoder/decoder depths 6/4, width 32, four heads, MLP ratio 4; 600 epochs, $B = 64$, mask ratio 0.5, 40 warmup epochs, base LR 10^{-3}, minimum LR 10^{-5}.
TabCSDI	Four diffusion layers, 64 channels, four heads; 50 quadratic-schedule steps; 250 epochs, LR 5×10^{-4}, $B = 64$.
DiffPuter	Hidden width 1,024; ten EM rounds, 20 internal samples, 50 diffusion steps; at most 10,000 epochs per M-step, LR 10^{-4}, $B = 4096$, training-loss patience 500; warm start.
TabImpute	V2 pretrained weights; context cap 2,048 rows; test-row batch 256; 5,000 distribution bins; one point completion.
Prediction specialists	
Ridge / logistic regression	Ridge penalty $\alpha = 1$ / inverse regularization $C = 1$; logistic regression uses at most 1,000 solver iterations.
Random forest	300 trees, minimum leaf size 1.
XGBoost	500 estimators, depth 6, LR 0.1; row/column fractions 1; early-stopping patience 50.
LightGBM	500 estimators, 31 leaves, LR 0.05, minimum child size 20; early-stopping patience 50.
CatBoost	500 iterations, depth 6, LR 0.05, leaf $L_2 = 3$; early-stopping patience 50.
MLP	ReLU widths $[256, 128]$, $L_2 = 10^{-4}$, initial LR 10^{-3}; at most 500 iterations, automatic batch size, patience 20.
FT-Transformer / TabM	Three blocks; 100 epochs, $B = 256$, patience 16. FT-Transformer: LR 10^{-4}, WD 10^{-5}. TabM: width 512, 32 ensemble members, dropout 0.1, LR 2×10^{-3}, WD 0.
CARD	1,000 linear-schedule diffusion steps; backbone width 128; guidance widths $[100, 50]$; guidance/diffusion budgets 1,000 epochs, LR 10^{-3}; diffusion $B = 32$, gradient clip 1; guidance patience 50.
NGBoost	Normal predictive distribution; 500 estimators, LR 0.01, full minibatches, early-stopping patience 50.
LightGBM-Quantile	500 estimators, 31 leaves, LR 0.05; quantiles $\{.025, .05, .1, .25, .5, .75, .9, .95, .975\}$; early-stopping patience 50.
Pretrained generalists	
TabPFN-3 + unsupervised	V3 default classifier/regressor checkpoints; one estimator/permutation; context cap 50,000; generation temperature 1 and imputation temperature 10^{-9}; cached fitting.
TabICLv2	Classifier/regressor checkpoints dated 2026-02-12; eight estimators, batch 8, context cap 50,000; imputation temperature 1 and two iterations.

Table 14 (continued)

Method	Configuration
LimiX-2	`LimiX-2.ckpt`; v2 no-retrieval classification, regression, and MVI configurations; context cap 50,000; prediction temperatures 0.9 (classification) and 1 (regression).

Resource allocation. The sharded launcher allocates one independent method–dataset worker to each visible GPU and balances datasets by $(n_{\text{train}} + n_{\text{test}})p$. Its OpenMP, MKL, and OpenBLAS thread defaults are two per worker.

H.2 Controlled-transfer implementation

The controlled study uses the five OpenML datasets in Table 15, training seeds $\{0, 1, 2\}$, five backbones, and seven objective arms $\{G, I, P, GI, GP, IP, GIP\}$. Axis scores follow the member-complete construction in Appendix E.

Table 15: Controlled-transfer datasets and train–test splits. Feature counts exclude the target. C and R denote classification and regression.

Dataset	OpenML ID	Type	Features	Train	Test
JapaneseVowels	976	C	14	7,968	1,993
MiamiHousing2016	43093	R	16	11,145	2,787
cleveland	786	C	13	242	61
credit_card_clients_default	46919	C	23	24,000	6,000
website_phishing	46963	C	9	1,082	271

Interventions and fitting. An enabled query type is sampled uniformly for each neural-training row, duplicated forest exposure, or Gaussian-copula optimizer update. G scores all variables; P observes the features and scores the target; I selects each feature for reconstruction with probability 0.3, enforces a nonempty query, and hides the target without scoring it. Observed, non-query variables form the conditioning inputs. The neural losses are masked to the queried coordinates; the forest objective sets unqueried gradients and Hessians to zero. Configurations in Table 16 are fixed within each backbone across datasets and arms. The three vendored neural pipelines use export protocol v6: a deterministic 10% internal holdout is drawn from the outer training split using seed 0, and training-time validation uses this holdout. TabSyn shares one objective-independent VAE per dataset–seed block across all seven arms.

TabSyn's shared VAE selects the checkpoint with the lowest categorical cross-entropy on the internal validation set, uses ReduceLROnPlateau (factor 0.95, patience 10), and initializes $\beta = 10^{-2}$ with decay factor 0.7 and threshold 10^{-5}. The causal branches bypass TabDiff's configured plateau LR scheduler and TabSyn's diffusion plateau scheduler and early stopping. TabDiff's configured `steps=8000` denotes epochs; TabDDPM's `steps=100000` denotes optimizer updates.

Evaluation draws and analysis. Each checkpoint is reused for all three tasks. Generation requests as many synthetic rows as the outer training split, with evaluation seed equal to the training seed. Imputation uses MCAR, MAR, MNAR, and their mixed mechanism at rates $\{0.1, 0.3, 0.5\}$, three draws per setting, and hidden target access; selection, masking, and imputation seeds equal the training seed. Prediction uses three conditional draws for Forest-VP Joint, TabDDPM, TabDiff, and TabSyn. Gaussian Copula uses native Monte Carlo budgets of 100 draws for regression and 200 for classification. Conditional sampling and training randomization are specified separately. TabDiff and TabDDPM use one imputation resampling round; TabSyn uses 50 EDM steps and 20 resampling rounds; Forest-VP uses RePaint with $r = 5$ and $j = 0.1$.

Uncertainty is estimated with 2,000 bootstrap draws and base random seed 20260924; 95% intervals are the 2.5th and 97.5th percentiles. Point estimates average across settings within each seed block, then across seeds within each backbone–dataset pair, and finally across those pair means. Wilcoxon diagnostics use the backbone–dataset means, with Holm adjustment within each analysis family.

Table 16: Training configurations for the controlled-transfer study. Budgets are backbone-specific and matched across objective arms within a dataset–seed block. LR denotes learning rate; WD denotes weight decay.

Backbone	Architecture and loss	Optimizer and configured budget
Gaussian Copula	Empirical marginals with at most 1,000 stored quantiles; joint/conditional Gaussian likelihood for the selected query.	Adam, LR 0.03, batch 1,024, 2,000 updates; gradient norm clipped at 10.
Forest-VP Joint	50 diffusion times; 100 duplicated exposures; query-masked multi-output squared-error trees.	100 boosting rounds per time, depth 7, shrinkage 0.3, histogram trees; one job and one XGBoost thread per worker.
TabDDPM	MLP widths $(1024, 2048, 2048, 1024)$, time embedding 128, no dropout; 1,000 diffusion times, cosine noise schedule; masked numerical MSE plus multinomial loss.	AdamW, LR 10^{-3} with linear decay, WD 5×10^{-4}, batch 4,096, 100,000 updates.
TabDiff	Two layers, token width 4, one head, factor 32, time width 1,024; learnable power-mean numerical and log-linear categorical schedules. Discrete loss weight 1; continuous weight $1 - e/8000$ at epoch e.	AdamW, fixed LR 10^{-3}, WD 0, batch 4,096, 8,000 epochs; EMA 0.997 and final EMA checkpoint; no separate target-only guidance model.
TabSyn	Shared VAE: two layers, token width 4, one head, factor 32; reconstruction plus β-KL. Latent EDM: width 1,024 and query-masked loss.	Adam, LR 10^{-3}, WD 0, batch 4,096; VAE 4,000 epochs; latent diffusion 10,001 epochs at fixed LR, saving the final model.

H.3 Computational environment

We use NVIDIA GeForce RTX 5090 GPUs with 32 GB of device memory per GPU. Task- and model-specific dependency lists are stored under `requirements/` in each code tree.