INCEPT: INFUSING NOVEL CONCEPTS FOR EXPLAINING PRETRAINED TABULAR MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

The Platonic Representation Hypothesis suggests that sufficiently capable models converge to similar internal representations regardless of architecture or training. Using embeddings from the TabArena benchmark to assess geometric convergence of tabular foundation models, we find transformer-based tabular foundation models (TFMs) cluster tightly, with hypernetworks and LLM-based models sitting apart. To explain the predictive gap between two TFMs, we train a sparse autoencoder (SAE) per model and identify their unmatched concepts. Two interventions are tested: *ablation* zeros unmatched concepts out of the stronger model's embedding; *transfer* maps them into the weaker model's embedding. Across 15 model pairs and 51 TabArena datasets, ablation closes 93% of the stronger model's per-row advantage, and transfer recovers 90% of the gap. The transferred concepts lie roughly half off the weaker model's existing manifold, revealing structure the recipient does not use but can absorb.

1 INTRODUCTION

Tabular foundation models (TFMs) have emerged as a promising approach to few-shot learning on structured data. Unlike traditional supervised learning, which requires training a new model for each dataset, TFMs are pretrained once and can generalize to new tabular tasks with minimal examples. The field has matured rapidly: a recent comprehensive survey (Jiang et al., 2025) organizes tabular representation learning into specialized, transferable, and general (foundation) models, reflecting the growing diversity of approaches. Yet these models differ dramatically in their design choices: TabPFN (Hollmann et al., 2023) learns from synthetic data sampled from Bayesian priors, CARTE (Kim et al., 2024) pretrains on real-world knowledge graphs, TabDPT (Ma et al., 2025) combines real data with retrieval augmentation, and Tabula-8B (Gardner et al., 2024) finetunes an LLM on serialized tables.

Do these diverse approaches converge to similar internal representations? The Platonic Representation Hypothesis (Huh et al., 2024) proposes that sufficiently capable models trained on related tasks will approximate the same underlying representation space, but this work focuses primarily on vision and language models, where architectures have largely converged on transformers trained on internet-scale data. Tabular foundation models present a more challenging test: architectures span transformers, hypernetworks, and graph neural networks; training data ranges from synthetic priors to real tables; and there is no shared "Common Crawl" equivalent.

Building on geometric tools—CKA and Procrustes alignment—we go beyond that comparison with concept-level causal analysis, examining what models learn and why they differ in performance. Using Sparse Autoencoders (SAEs), we decompose each model's representations into interpretable concept dictionaries, match concepts across models, and test their functional relevance through causal interventions: ablating unique concepts to degrade predictions, and transferring them into a weaker model to close its per-row gap. Our contributions are summarized as follows:

- We provide the first geometric comparison of internal representations across eight diverse tabular foundation models, spanning synthetic-data-trained (TabPFN, Mitra, TabICL, TabICL-v2), real-data-trained (TabDPT, CARTE), hypernetwork (HyperFast), and LLM-based (Tabula-8B) architectures.

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.

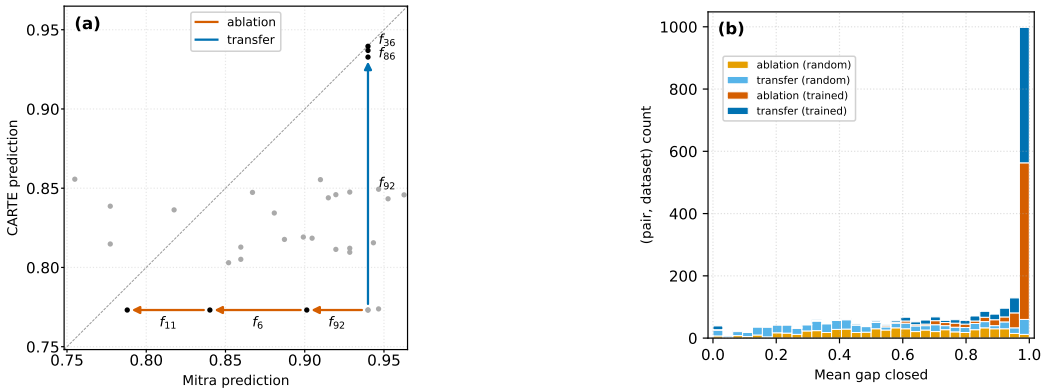


Figure 1: **(a)** Illustration of our two intervention methods for a single row from credit-g (CARTE vs. Mitra): gray dots show test point predictions in the highlighted range; ablation removes Mitra's unmatched concepts f_{92}, f_6, and f_{11} (red leftward arrows), closing the gap by degrading its prediction toward CARTE's; transfer injects f_{92}, f_{86}, and f_{36} into CARTE (blue upward arrows), making CARTE's prediction as confident as Mitra's. **(b)** Distribution of mean gap closed (gc) across all (dataset, model pair) experiments. Trained concepts (dark) concentrate near $gc = 1.0$ while random baselines (light) are broadly distributed, the trained–random difference is the learned signal. Intervention details in Algorithm 1. Pipeline overview in Appendix 4.

- We introduce a causal methodology for explaining performance disparities between tabular foundation models, showing that ablating 2–5 unmatched SAE concepts closes 93% of the prediction gap across 15 model pairs (six architecturally comparable models) and 51 datasets.
- We demonstrate cross-model concept transfer via global concept maps, showing that the weaker model's gap is recoverable: injecting a median of 3–6 missing concepts closes 90% of it across 15 model pairs and 51 datasets.
- We release our SAE training code, learned dictionaries, and analysis tools to support future work on tabular foundation model interpretability.

2 Related Work

We build on three lines of research: tabular foundation models, theoretical work on representation convergence, and sparse autoencoders for interpretability. For a comprehensive overview of tabular representation learning, we refer readers to Jiang et al. (2025).

Tabular foundation models Recent years have seen rapid development of foundation models for tabular data. These models aim to perform few-shot predictions on new datasets without task-specific training. TabPFN (Hollmann et al., 2023) pioneered this approach by training a transformer to approximate Bayesian inference on synthetic datasets sampled from structural causal model priors. TabPFN v2 (Hollmann et al., 2025) extends this to larger datasets and regression tasks. Mitra (Zhang et al., 2025) combines causal structure priors with gradient-boosted tree inductive biases, using a novel 2D attention mechanism. TabICL (Qu et al., 2025) scales synthetic pretraining to handle datasets with up to 500K samples. An alternative approach pretrains on large corpora of real tabular datasets. TabDPT (Ma et al., 2025) combines OpenML datasets with self-supervised learning and retrieval augmentation. CARTE (Kim et al., 2024) pretrains a graph transformer on the YAGO3 knowledge base, learning entity and relation semantics. XTab (Zhu et al., 2023) and UniTabE (Yang et al., 2024) explore cross-table pretraining strategies. TabularFM (Tran et al., 2024) scales to over 1 million tables from Kaggle and GitTables.

Several works adapt large language models (LLMs) to tabular data via serialization. Tabula-8B (Gardner et al., 2024) is a fine-tuned version of Llama 3-8B using T4, a corpus of 3.1 million tables (1.34TB uncompressed) derived from TabLib (Eggert et al., 2023), representing the largest-scale tabular pretraining effort to date. TabLLM (Hegselmann et al., 2023) and LIFT (Dinh et al.,

2022) explore different serialization and prompting strategies for frozen or finetuned LLMs. MotherNet (Müller et al., 2023) and HyperFast (Bonet et al., 2024) take a different approach: rather than processing data directly, they generate the weights of a task-specific neural network conditioned on the training set.

Representation convergence Platonic Representation Hypothesis (Huh et al., 2024) proposes that models trained on different data and objectives converge to similar representations, up to simple transformations. They provide evidence from vision and language models, showing that representation similarity increases with model capability. Complementing work on representation convergence, Kaushik et al. (2025) demonstrates that neural networks converge to shared low-dimensional subspaces in weight space. These two hypotheses operate at different levels of abstraction: weight space (the parameters) vs. representation space (the activations). Recent work has questioned whether raw CKA values reliably measure convergence, identifying width and depth confounders that inflate similarity scores (Gröger et al., 2026). We complement geometric comparison with SAE-based causal analysis: unmatched features reflect structural gaps, confirmed by ablation and transfer rather than geometric alignment alone.

Standard geometric tools — CKA (Kornblith et al., 2019), SVCCA (Raghu et al., 2017), and Procrustes alignment (Schönemann, 1966) — compare embedding geometry but cannot identify the reasons for divergence. Our work introduces a third level: *concept space*—the discrete, interpretable concepts that can be extracted from activations via sparse dictionary learning. Concept-level comparison offers advantages over geometric methods: it can identify which specific patterns are shared or distinct, not just whether spaces align overall.

Sparse autoencoders for interpretability Sparse autoencoders learn overcomplete dictionaries of interpretable concepts from neural network activations (Bricken et al., 2023; Cunningham et al., 2024). Each dictionary element activates on semantically coherent subsets of inputs. Standard SAEs suffer from instability: different training runs produce different dictionaries (Bricken et al., 2023). Archetypal SAEs (Fel et al., 2025) address this by constraining dictionary elements to the convex hull of the data, providing stability guarantees. Matryoshka SAEs (Bussmann et al., 2025) learn nested dictionaries at multiple granularity levels, with smaller dictionaries capturing general concepts and larger ones capturing specific details. This prevents "concept absorption" where general concepts are subsumed into specific ones.

SAE concepts have been used for targeted interventions: steering model behavior by activating or suppressing specific concepts (Templeton et al., 2024). While prior work has matched SAE features across LLMs (Lan et al., 2024) or vision models (Thasarathan et al., 2025), our approach—learning one SAE per model, then comparing dictionaries via causal interventions—is the first to apply this analysis to tabular foundation models, and the first to validate cross-model concept correspondence through targeted causal interventions rather than observational similarity alone.

Our empirical analysis builds on TabArena (Erickson et al., 2025), a comprehensive benchmark suite for tabular foundation models covering diverse dataset types and evaluation protocols.

3 Experimental Setup

Data TabArena spans multiple domains (healthcare, finance, e-commerce, scientific) with standardized preprocessing, evaluation protocols, and data splits. We use TabArena (Erickson et al., 2025) rather than larger pretraining corpora (e.g., T4 (Gardner et al., 2024)) because its curated domain labels enable stratified analysis and its benchmark setting ensures out-of-distribution evaluation for models pretrained on other corpora. For each model-dataset pair, we use two of three outer folds (or stratified subsamples for models not capable of using the full set) for context samples (train set) and run inference on the remaining fold as the query (test). TabICL v1 and HyperFast are classification-only, limiting their evaluation to the 38 classification datasets; all other models are evaluated on the full 51.

Tabular foundation models We study eight TFMs spanning three paradigms: synthetic-data-pretrained (TabPFN v2.5, Mitra, TabICL v1/v2), real-data-pretrained (TabDPT, CARTE), hypernetwork (HyperFast), and LLM-based (Tabula-8B) — full details in Appendix A. TabDPT addition-

ally performs per-query nearest-neighbour retrieval over the training split; we run it in this native retrieval-augmented mode throughout, including inside the interventions.

Embedding extraction In deep learning, the early layers typically refine the input primitives into increasingly abstract representations, while the later layers shift focus toward the task objective Zeiler & Fergus (2014) (Tenney et al., 2019). To find the layer where the optimal representations could be found for each model, we set output hooks at every intermediate layer to cache intermediate activations for every dataset. We use centered kernel alignment Kornblith et al. (2019), refer to equation 1, to analyze the extraction point, which identifies the inflection point where representational similarity drops as the model transitions toward task-specific processing, following recent work using CKA drift to localize critical layers (Liu et al., 2025; Gupta et al., 2026). We use the mean inflection point per task type as the extraction layer. Refer to Figure 6 in Appendix C for an example, and Table 5 for details. With the extraction layer fixed, we generate embeddings from the test fold as queries.

Geometric alignment We measure the similarity of the embeddings using standard geometric methods. Following Kornblith et al. (2019), given embeddings $X \in \mathbb{R}^{n \times d_x}$ and $Y \in \mathbb{R}^{n \times d_y}$ for the matched n examples, we compute linear Centered Kernel Alignment (CKA) as:

$$\mathrm{CKA}(X, Y) = \frac{\|\tilde{Y}^\top \tilde{X}\|_F^2}{\|\tilde{X}^\top \tilde{X}\|_F \, \|\tilde{Y}^\top \tilde{Y}\|_F}, \qquad \tilde{X} = X - \mathbf{1}\bar{x}^\top, \quad \tilde{Y} = Y - \mathbf{1}\bar{y}^\top \tag{1}$$

CKA is invariant to isotropic scaling and orthogonal transformation, providing a measure of whether two embeddings encode a similar structure. We additionally compute orthogonal Procrustes disparity (Schönemann, 1966) as a confirming geometric check (Appendix C.11).

3.1 Sparse Autoencoders

Training corpus We standardize embeddings per dataset and construct the SAE training corpus via stratified sampling by class label and prediction difficulty (terciles defined by TabPFN v2.5 error), capping at 700 samples per dataset (500 train, 200 held out from SAE training for the causal interventions) to maintain approximately equal dataset representation.

Training We train one SAE per model on concatenated TabArena activations using a combination of Matryoshka (Bussmann et al., 2025), archetypal (Fel et al., 2025), and top-k (Gao et al., 2025) objectives: Matryoshka mitigates concept splitting, archetypal constraints improve cross-seed stability, and top-k sparsity reinforces both. We select SAE hyperparameters via an Optuna (Akiba et al., 2019) TPE sweep (30 trials per model, full search space in Appendix D), optimizing $J = \ell_{\text{recon}}^{\text{test}} \cdot \sqrt{d_{\text{hidden}}} \cdot \sqrt{L_0} \,/\, \rho_{\text{alive}}$, where L_0 is the average number of active latents per input and ρ_{alive} is the fraction of latents that fire on any input—balancing reconstruction quality against dictionary size and sparsity while penalizing dead features. Among the sweep configurations that satisfy $R^2 \geq 0.80$, alive fraction ≥ 0.80, and cross-seed stability ≥ 0.75, we select the least complex: the smallest $\sqrt{d_{\text{hidden}} L_0}$ (the two capacity terms of J); architecture and sweep details are in Appendix D. As a geometry-matched control used throughout, we build a random-SAE baseline from each trained SAE that keeps the data-derived archetype dictionary but re-initializes the concept assignments (archetype logits and deviations) at random, so it shares the trained SAE's data-aligned geometry and differs only in which concepts form.

3.2 Concept-Level Matching across Dictionaries

The CKA analysis establishes that models share geometric structure, but does not reveal what that structure encodes or why models differ in performance. To answer these questions, we decompose each model's representations into interpretable concept dictionaries using sparse autoencoders, then identify which concepts correspond across models and which are unique to each.

Activation-based similarity Since models have different embedding dimensions (192 to 768), their decoder weight vectors are not directly comparable. Instead, we compare concepts based

on their activation patterns across shared data. For each pair of models, we compute the Pearson correlation between SAE concept activations across all shared TabArena test rows:

$$\text{act-sim}(i, j) = \text{corr}\left([f_i^{(A)}(x)]_{x\in\mathcal{X}}, [f_j^{(B)}(x)]_{x\in\mathcal{X}}\right) \tag{2}$$

where $\mathcal{X}$ is the set of shared TabArena test rows. Concepts from different SAEs are considered corresponding if they fire on the same rows.

Mutual nearest neighbors The first stage of matching identifies cross-model concept correspondences using mutual nearest neighbors (MNN) Dekel et al. (2015); Conneau et al. (2018) on the activation-correlation matrix (Eq. 2), following the activation-similarity matching strategy used in cross-model SAE analyses of LLMs (Lan et al., 2024). A match is established when concept i in model A and concept j in model B are each other's nearest neighbor in the cross-correlation matrix — that is, j has the highest |act-sim| with i among all of B's concepts, and i has the highest correlation with j among all of A's concepts. The mutual constraint filters out spurious correlations that are strong in one direction but not reciprocated. This is particularly important given known failure modes in SAE matching: concept absorption (Chanin et al., 2025b), hedging (Chanin et al., 2025a), and polysemanticity can all produce false correspondences between independently trained dictionaries. We set the minimum correlation threshold using random SAE baselines, establishing a noise floor below which concepts are considered unmatched.

Community detection Assembling all pairwise MNN matches across models into a single graph produces large connected components. Without further partitioning, transitively linked concepts — only indirectly related — would be grouped together. We apply Leiden community detection Traag et al. (2019) to partition this graph into tightly-connected communities, each representing a single shared concept. Edge weights are the activation correlations from Equation 2, so the algorithm groups concepts that are mutually similar rather than merely transitively linked. Graph-based grouping of related SAE features has been used previously to discover hierarchical "feature families" within a single model (O'Neill et al., 2024); here we extend it to grouping across models.

3.3 Intervention

The concept matching pipeline identifies unmatched concepts for any two models, strong (S) and weak (W), on any row from a dataset. For each pair, S is the dataset-level stronger model (AUC for classification, RMSE for regression)[1], and interventions are applied per row, restricted to rows where S's per-row loss is strictly lower. Both interventions steer the recipient toward S's output, so S's per-row prediction is a required input to the procedure rather than an inferred quantity. We calculate an unmatched concept importance matrix I for each row via single-concept ablation, then test those rank-ordered concepts via causal interventions, ablation and transfer (Algorithm 1), to determine whether they explain the predictive difference. The number of accepted concepts K is not preset: the per-row greedy admits concepts in importance order while each moves the recipient toward S, and stops when gap closed reaches a tolerance or no remaining concept improves it. The two causal interventions share a common backbone: extract test-row activations from each model, encode through that model's SAE, manipulate one concept at a time, decode, and re-run the model's layers downstream of the extraction point with the modified hidden state to produce an updated prediction. Ablation removes a strong model's unmatched concepts; transfer injects them into a weaker model after mapping through a cross-space ridge fit.

Transfer requires a map between the two models' embedding spaces. We estimate a global linear map $M \in \mathbb{R}^{d_{\text{weak}} \times d_{\text{strong}}}$ from matched concept pairs via ridge regression, filtering landmark correlations above the random-SAE noise floor (details in Appendix F.8). For both interventions, we report *gap closed*: the fraction of the per-row prediction gap eliminated by the selected concepts — near 1.0 indicates the advantage is fully explained by unique SAE concepts, near 0 indicates the

[1] On four low-event-rate classification datasets (seismic-bumps, hiva_agnostic, taiwanese_bankruptcy_prediction, Marketing_Campaign) the query holdout contains a single class, leaving AUC undefined; there we fall back to negative mean cross-entropy, which is still defined and ranks the models by calibration. The fallback is applied identically in the trained and random arms, and no dataset is dropped.

Algorithm 1 Cross-Model Concept Intervention

Require: Strong model S, weak model W, unmatched concept pool C, importance matrix I, mode $\in \{\text{ablation}, \text{transfer}\}$
Ensure: Selected concepts and gap closed gc

```
selected ← []
for each test row x where S outperforms W do
    rank C by per-row importance I(x)
    for c in ranked C do
        if mode == ablation then
            ablate c from S's embedding
            if row-level distance to W decreases then selected.APPEND(c)
        else if mode == transfer then
            map c via ridge regression M into W's embedding space
            inject mapped direction into W's hidden state
            if W's prediction improves then selected.APPEND(c)
return selected, mean gap closed across all rows
```

unmatched concepts did not account for the advantage. Figure 1 illustrates both interventions for a single credit-g row. To interpret concepts, we apply a third, input-side intervention: the smallest edit from donor non-activating rows that drops a concept's activation with minimal effect on the weak model's prediction (Appendix F.3).

4 Results

4.1 Geometric Alignment

Figure 2 reports pairwise CKA similarity across the eight models on TabArena classification datasets. A tight transformer cluster emerges (Mitra, TabICL, TabICL-v2, TabDPT; CKA 0.76–0.82); TabPFN sits in an intermediate position adjacent to the cluster. HyperFast and Tabula-8B are geometrically the most distant and are excluded from the causal analyses in §4.2. The CKA values for the regression datasets are available in Table 7.

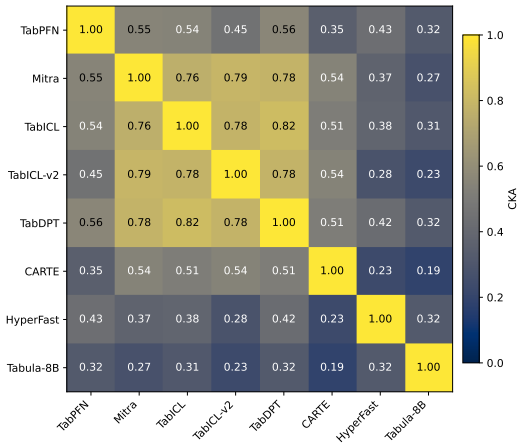


Figure 2: Linear CKA similarity ($\in [0, 1]$) between model representations, averaged over TabArena classification datasets. Darker = lower similarity, lighter = higher.

These similarities are notable given the architectural diversity of the models studied; Procrustes disparity (Appendix 8) confirms the same clustering. The geometric distinctness of HyperFast and Tabula-8B, combined with architectural differences in representation formation (HyperFast) and context size constraints (Tabula-8B), precluded further investigation within this work.

4.2 CONCEPT INTERVENTION

Concept importance is not uniform; a small number of high-importance directions dominate each model's predictive advantage, varying by intervention and task type. Figure 1 illustrates both interventions for a single row from credit-g (CARTE vs. Mitra), showing how ablation and transfer select different, but overlapping, subsets of unmatched concepts. The concept f_{92} — sensitive to credit history status — appears in both interventions (-0.04 in ablation, $+0.16$ in transfer), identifying it as the primary structural gap between the two models. Ablation additionally removes f_6 and f_{11}, sensitive to loan duration and checking account balance range respectively; transfer injects f_{86} and f_{36}, both sensitive to checking account status and loan duration, suggesting CARTE lacks sensitivity to these credit risk indicators that Mitra detects. Input-level patches for each concept are reported in Appendix 15.

Figure 3 shows ablation and transfer interventions (Mitra to CARTE) across three task types: binary classification (credit-g), multi-class classification (students_dropout_and_academic_success), and regression (houses). Gray dots are baseline predictions; each point sprouts two line segments from its baseline. Ablation modifies the strong model's prediction, so post-ablation points move horizontally; transfer modifies the weak model's prediction, so post-transfer points move vertically. Full results for all 51 datasets appear in Appendix I.

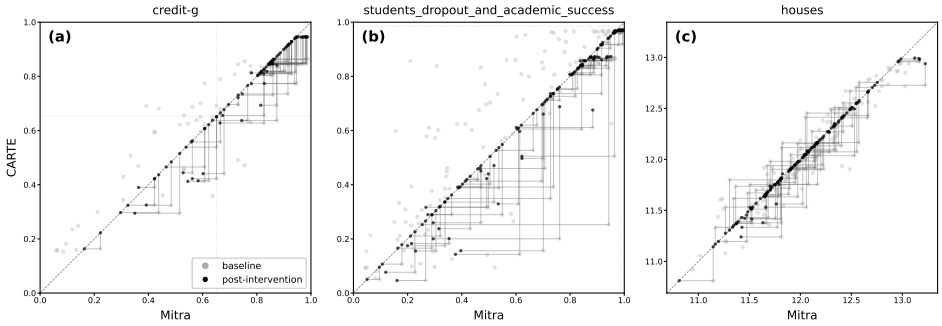


Figure 3: Ablation (horizontal lines) and transfer (vertical lines) for Mitra (strong) vs. CARTE across binary, multi-class, and regression tasks. Gray: baseline predictions. Black: post-intervention.

Input-level patches confirm concept specificity across all three datasets: Mitra uniquely detects credit risk indicators (foreign worker status, employment tenure) in credit-g, academic trajectory patterns (approved curricular units, age at enrollment) in students_dropout, and geographic pricing signals (latitude, longitude) in houses — Appendix 17–19.

Ablation Table 1 reports the mean gap closed across all 15 pairs and 699 comparisons[2]: unique SAE concepts explain on average 93% of the strong model's per-row advantage. The pairs with highest gap closed have TabICL-v2 (0.99) and TabPFN (0.98) as the strong model, while Mitra (0.74) and CARTE (0.72) share the lowest gap closed when they are strong (CARTE is strong on only 16 comparisons vs. Mitra's 88). TabPFN distinguishes itself by the largest net learned signal over the random baseline ($\Delta = 0.51$), meaning its unmatched concepts are substantially more effective than random directions at closing the gap. Ablation is selective: it accepts only about 17% of the concepts it tries (Table 1).

Transfer The ablation experiments demonstrate that removing a model's unique concepts degrades its predictions. The complementary question is whether *adding* those concepts to a weaker model improves it. For each model pair and dataset where the strong model outperforms the weak, we map the strong model's unmatched decoder directions into the weak model's embedding space via a global ridge regression fitted on all matched direction pairs, and inject the mapped directions into the weak model's hidden state.

[2]Of 15×51=765 possible (pair, dataset) comparisons, 65 are removed by TabICL's classification-only constraint and 1 (QSAR-TID-11 with Mitra as weak) by GPU memory limits, leaving 699.

Table 1: Ablation results by strong model, sorted by N. gc/gc_R = mean gap closed using trained/random SAEs. K/K_R = mean concepts ablated using trained/random SAEs. *acc* = concepts ablated / concepts tried, where *tried* is the rank depth reached in the importance-ordered candidate list (the position of the deepest accepted concept), since the greedy tries concepts in importance order and stops at the gap-closure tolerance.

Model (when strong)	N	gc	gc_R	K	K_R	acc
TabICL-v2	185	0.99 ± 0.07	0.73 ± 0.22	2.2 ± 2.8	11.6 ± 6.4	0.142
TabPFN	149	0.98 ± 0.09	0.47 ± 0.23	2.9 ± 5.0	11.6 ± 6.1	0.166
TabDPT	137	0.94 ± 0.15	0.53 ± 0.21	3.4 ± 2.7	6.1 ± 3.5	0.232
TabICL	124	0.96 ± 0.11	0.64 ± 0.20	1.9 ± 3.3	14.5 ± 7.4	0.125
Mitra	88	0.74 ± 0.17	0.46 ± 0.22	2.3 ± 1.4	3.1 ± 3.5	0.324
CARTE	16	0.72 ± 0.30	0.40 ± 0.21	4.9 ± 3.9	9.3 ± 5.2	0.228
Overall	699	0.93 ± 0.15	0.58 ± 0.24	2.6 ± 3.4	9.9 ± 6.9	0.174

Table 2: Transfer results by weak model, sorted by N. gc/gc_R = mean gap closed using trained/random SAEs. K/K_R = mean concepts injected using trained/random SAEs. *acc* = concepts accepted / concepts tried, where *tried* is the rank depth reached in the donor's importance-ordered candidate list (the position of the deepest accepted concept).

Model (when weak)	N	gc	gc_R	K	K_R	acc
CARTE	226	0.73 ± 0.27	0.28 ± 0.19	17.6 ± 20.9	14.4 ± 12.9	0.424
Mitra	140	0.99 ± 0.03	0.69 ± 0.19	4.0 ± 1.8	21.0 ± 10.2	0.382
TabDPT	105	0.98 ± 0.04	0.58 ± 0.30	6.0 ± 13.7	15.0 ± 9.6	0.360
TabPFN	91	0.98 ± 0.06	0.59 ± 0.23	6.3 ± 9.3	23.4 ± 12.5	0.457
TabICL	66	0.95 ± 0.11	0.54 ± 0.28	3.2 ± 1.5	16.1 ± 16.3	0.391
TabICL-v2	59	1.00 ± 0.01	0.86 ± 0.19	3.1 ± 1.7	16.7 ± 12.1	0.371
Overall	687	0.90 ± 0.20	0.52 ± 0.30	8.9 ± 14.9	17.3 ± 12.6	0.412

Across 687 (pair, dataset) experiments[3], transfer achieves a mean gap closed of 0.90 (median 0.99); over 71,430 strong-win rows pooled, 85% close more than 90% of the gap. Within the transformer cluster, per-pair transfer ranges from 0.93 to 1.00; pairs with CARTE as the recipient achieve the lowest transfer (0.67–0.79), consistent with CARTE's geometric distance from the transformer cluster. Transfer relies on more concepts than ablation: it accepts about 41% of the concepts it tries (Table 2) and needs more of them ($K \approx 8.9$ vs. ablation's 2.6) to close a comparable gap, because each injected concept carries a smaller effect — the median donor-importance of an accepted transfer concept is 0.005, against 0.033 for the concepts ablation removes.

Decomposing each injected update Δ (the net displacement from all accepted concepts on a row) against the recipient's active manifold (its top principal subspace) shows the injection is neither purely aligned nor purely novel: a median 0.45–0.64 of Δ's energy lies off the recipient's existing manifold across the six recipients (median principal angle 27–66°; Table 3), the rest overlapping structure the recipient already encodes. That the weak model productively absorbs a substantially off-manifold injection (closing $gc = 0.90$ with a median of 3–6 concepts) indicates the two models share exploitable structure that a single similarity score does not resolve. In 27.6% of comparisons, transfer gap closed exceeds ablation gap closed, consistent with the ridge map underestimating injection magnitude relative to greedy ablation. We causally patch these off-manifold concepts at the input level in Appendix F.9, tying them to interpretable column edits.

Because Tables 1 and 2 intervene only on below-diagonal rows, those where the overall-stronger model wins per row, we check that the effect is not an artifact of this selection by running both interventions on the complementary above-diagonal rows, where the *nominally weaker* model wins per row and which are otherwise left untreated. Between them the two triangles cover 99.96% of

[3] Of the 699 ablation comparisons (Table 1), 13 fail in the transfer pipeline (5 Bioresponse, 5 hiva_agnostic, 3 QSAR-TID-11) due to map-fitting or memory failures, and 1 (carte_vs_mitra QSAR-TID-11) succeeds in transfer despite failing in ablation, leaving 687.

Table 3: Subspace decomposition of the transfer injection, per recipient. *Off-manifold energy* is the median fraction of each accepted transfer delta lying outside the recipient's active manifold (its top principal subspace); *principal angle* is the median angle between the delta and that subspace ($0°$ = aligned, $90°$ = orthogonal); K is the median number of concepts transferred. Medians are over the below-diagonal accepted transfers.

Recipient	Off-manifold energy	Median principal angle	Median K
TabPFN	0.49	$48°$	5
TabICL	0.50	$52°$	4
TabICL-v2	0.64	$66°$	3
Mitra	0.60	$54°$	4
TabDPT	0.53	$58°$	3
CARTE	0.45	$27°$	6

the transfer test set and 99.19% of the ablation test set.[4] The effect holds on the newly-treated rows: over the $46{,}446$ above-diagonal transfer rows, injecting the weaker model's unmatched concepts into the stronger one (weak→strong) closes a pooled 0.98 of the per-row gap, and over the $48{,}237$ above-diagonal ablation rows, ablation closes a pooled 0.89. Knowing which model is globally stronger is thus an experimental control for the causal test, not a precondition for the intervention's effect.

Transfer also never degrades the recipient; this follows from the greedy acceptance rule, which adds a concept only if it moves the recipient's prediction toward the lower-loss donor. Per-pair values vary around these means: ablation is weakest when Mitra is the strong model and transfer when CARTE is the recipient (Appendix F.8, Figure 20); ablation selects 2–5 concepts per row and transfer a median of 3–6, since per-concept importance decays steeply (Appendix F.2).

Limitations The interventions are a causal diagnostic, not a deployment method: they take per-row knowledge of which model is stronger as an experimental control. Random-SAE directions of matched norm close 52% of the transfer gap against 90% for trained concepts (Appendix G), and all closures are measured on a fixed subsample of up to 200 test rows per dataset. The transfer map is a global linear fit that explains little variance (negative cross-validated R^2, Appendix F.8); the transfer evidence rests on the interventions and the random control, not on map fidelity. The causal results cover six models: HyperFast is excluded because it predicts in weight space, and Tabula-8B because a 32-row context leaves its predictions unreliable on TabArena's mostly imbalanced tasks (Appendix C.8, C.9). Newer TFMs are further pairs for the same diagnostic rather than a change to it; six models across three families suffice to establish that the gap is concept-structured. Concept semantics rest on patch examples (Appendix F.3) rather than validation at scale; further caveats (greedy search, magnitude bias, map quality, benchmark scope) are in Appendix F.10.

5 Conclusion

We investigated whether tabular foundation models converge to similar internal representations. CKA analysis reveals a tight transformer cluster (0.76–0.82) despite radically different architectures, with 846 multi-model concept groups shared across models — 45 universal across all six. Ablating unmatched concepts closes 93% of the per-row performance gap within 2–5 concepts, confirming that divergence is concept-structured rather than architectural. Transferring these concepts into weaker models closes their per-row gap, with transfer success correlating with geometric proximity — evidence of a shared representational space each model only partially exploits: a median 0.45–0.64 of each transferred update lies off the recipient's existing manifold. These interventions are a causal diagnostic of *why* models differ, not a deployment method: because they take per-row knowledge of the stronger model as an experimental control, the concept-level map of what each model knows that others do not is a basis for understanding these differences and guiding future architecture design, not a route to costless accuracy gains. Geometry tells you the distance; concepts give you a compass and a map.

[4] The remainder is tied rows plus the above-diagonal rows of the 9 comparisons whose reverse sweep hit GPU memory limits.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods, clean and reformat datasets, support qualitative and thematic data analysis (labeling and grouping learned concepts), interpret results, design and provide feedback on the experimental methodology, and refine hypotheses. We have not used generative AI tools to generate synthetic datasets or to develop the conceptual framework of the work; translation and mathematical proofs are not applicable to this work. Additionally, we used generative AI tools to edit and refactor software code, format references, draft and edit parts of the manuscript for readability, suggest document structure, brainstorm, and search for and summarize related literature. We have reviewed all AI-assisted work: AI-generated code was reviewed and tested by the authors, and every reported number is produced by that code from recorded intervention outputs; AI-assisted interpretations were checked against the underlying quantitative evidence; and concept labels were validated against the activation patterns and input examples they summarize. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All experiments use the publicly available TabArena benchmark datasets and publicly released model checkpoints. Full experimental details are provided in the appendices, which document the tabular foundation models used, sparse autoencoder training, SAE concept construction and labeling, the causal intervention and ablation protocol, and the random and observational baselines. The code for every stage — embedding extraction, SAE training, concept matching, the interventions, the decomposition, and the patch search — is available anonymously at https://anonymous.4open.science/r/incept-anon-4DBC/; the trained SAE checkpoints, the matching files, and the per-row intervention outputs from which every table and figure is generated will be released with the code upon publication. The release pins the random seeds, the CUDA allocator configuration, and the model fitting paths (in particular TabDPT's ensemble and feature-projection draws), and each output records the host, GPU, and commit that produced it. Bit-exact reproduction holds on a given GPU architecture; across architectures (RTX 3090 vs. 4090) individual predictions differ at the 10^{-3} level and greedy intervention paths can diverge. A complete re-run of the pipeline under these controls reproduces the qualitative results reported here (for example, mean transfer gap closed 0.906 vs. the 0.898 reported, and the same median off-manifold energy), with drift in individual numbers; the values in this paper are those of the submission run.

ETHICS STATEMENT

This work studies the interpretability of tabular foundation models using publicly available benchmark datasets and introduces no new data collection involving human subjects. The authors identified no significant ethical concerns arising from this work.

REFERENCES

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2623–2631, 2019.

David Bonet, Daniel Mas Montserrat, Xavier Giró-i Nieto, and Alexander G. Ioannidis. HyperFast: Instant classification for tabular data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024. AAAI 2024.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023.

Bart Bussmann, Noa Nabeshima, Adam Karvonen, and Neel Nanda. Learning multi-level features with matryoshka sparse autoencoders. In *International Conference on Machine Learning (ICML)*, 2025.

David Chanin, Tomáš Dulka, and Adrià Garriga-Alonso. Feature hedging: Correlated features break narrow sparse autoencoders, 2025a. URL https://arxiv.org/abs/2505.11756.

David Chanin, James Wilken-Smith, Tomáš Dulka, Hardik Bhatnagar, Satvik Golechha, and Joseph Bloom. A is for absorption: Studying feature splitting and absorption in sparse autoencoders. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025b. URL https://arxiv.org/abs/2409.14507.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Word translation without parallel data. In *International Conference on Learning Representations (ICLR)*, 2018.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/1fa1ab11f4bd5f94b2ec20e794dbfa3b-Paper-Conference.pdf.

Tali Dekel, Shai Oron, Shai Avidan, Michael Rubinstein, and William T Freeman. Best-buddies similarity for robust template matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2026–2034, 2015.

Tuan Dinh, Yuchen Zeng, Ruisu Zhang, Ziqian Lin, Michael Gira, Shashank Rajput, Jy-yong Sohn, Dimitris Papailiopoulos, and Kangwook Lee. LIFT: Language-interfaced fine-tuning for non-language machine learning tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. NeurIPS 2022.

Gus Eggert, Kevin Huo, Mike Biven, and Justin Waugh. TabLib: A dataset of 627m tables with context. *arXiv preprint arXiv:2310.07875*, 2023.

Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. Autogluon-tabular: Robust and accurate automl for structured data, 2020. URL https://arxiv.org/abs/2003.06505.

Nick Erickson, Lennart Purucker, Andrej Tschalzev, David Holzmüller, Prateek Mutalik Desai, David Salinas, and Frank Hutter. TabArena: A living benchmark for machine learning on tabular data. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2025.

Thomas Fel, Ekdeep Singh Lubana, Jacob S. Prince, Matthew Kowal, Victor Boutin, Isabel Papadimitriou, Binxu Wang, Martin Wattenberg, Demba Ba, and Talia Konkle. Archetypal SAE: Adaptive and stable dictionary learning for concept extraction in large vision models. In *International Conference on Machine Learning (ICML)*, 2025.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/42ef3308c230942d223c411adf182c88-Paper-Conference.pdf.

Josh Gardner, Juan C. Perdomo, and Ludwig Schmidt. Large scale transfer learning for tabular data via language modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Fabian Gröger, Shuo Wen, and Maria Brbić. Revisiting the platonic representation hypothesis: An aristotelian view, 2026. URL https://arxiv.org/abs/2602.14486.

Aviral Gupta, Armaan Sethi, and Dhruv Kumar. TabPFN through the looking glass: An interpretability study of TabPFN and its internal representations, 2026. URL https://arxiv.org/abs/2601.08181.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. TabLLM: Few-shot classification of tabular data with large language models. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 206 of *Proceedings of Machine Learning Research*, pp. 5549–5581. PMLR, 2023. URL https://proceedings.mlr.press/v206/hegselmann23a.html.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. TabPFN: A transformer that solves small tabular classification problems in a second. In *International Conference on Learning Representations (ICLR)*, 2023.

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

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. The platonic representation hypothesis. In *International Conference on Machine Learning (ICML)*, 2024.

Jun-Peng Jiang, Si-Yang Liu, Hao-Run Cai, Qile Zhou, and Han-Jia Ye. Representation learning for tabular data: A comprehensive survey. *arXiv preprint arXiv:2504.16109*, 2025.

Prakhar Kaushik, Shravan Chaudhari, Ankit Vaidya, Rama Chellappa, and Alan Yuille. The universal weight subspace hypothesis. *arXiv preprint arXiv:2512.05117*, 2025.

Myung Jun Kim, Léo Grinsztajn, and Gaël Varoquaux. CARTE: Pretraining and transfer for tabular learning. In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2402.16785.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning (ICML)*, 2019.

Michael Lan, Philip Torr, Austin Meek, Ashkan Khakzar, David Krueger, and Fazl Barez. Quantifying feature space universality across large language models via sparse autoencoders, 2024. URL https://arxiv.org/abs/2410.06981.

Xuyuan Liu, Lei Hsiung, Yaoqing Yang, and Yujun Yan. Spectral insights into data-oblivious critical layers in large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025. URL https://arxiv.org/abs/2506.00382.

Junwei Ma, Valentin Thomas, Rasa Hosseinzadeh, Alex Labach, Hamidreza Kamkari, Jesse C. Cresswell, Keyvan Golestan, Guangwei Yu, Anthony L. Caterini, and Maksims Volkovs. TabDPT: Scaling tabular foundation models on real data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. Layer6 AI.

Andreas Müller, Carlo Curino, and Raghu Ramakrishnan. MotherNet: Fast training and inference via hyper-network transformers. *arXiv preprint arXiv:2312.08598*, 2023. Microsoft Research.

Charles O'Neill, Christine Ye, Kartheik Iyer, and John F. Wu. Disentangling dense embeddings with sparse autoencoders, 2024. URL https://arxiv.org/abs/2408.00657.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A tabular foundation model for in-context learning on large data. In *International Conference on Machine Learning (ICML)*, 2025. arXiv:2502.05564.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Peter H. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 1966.

Xiaoqing Sun, Alessandro Stolfo, Joshua Engels, Ben Wu, Senthooran Rajamanoharan, Mrinmaya Sachan, and Max Tegmark. Dense SAE latents are features, not bugs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C Daniel Freeman, Theodore R Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. *Anthropic Technical Report*, 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1452. URL https://aclanthology.org/P19-1452.

Harrish Thasarathan, Julian Forsyth, Thomas Fel, Matthew Kowal, and Konstantinos Derpanis. Universal sparse autoencoders: Interpretable cross-model concept alignment. In *International Conference on Machine Learning (ICML)*, 2025. URL https://arxiv.org/abs/2502.03714.

V. A. Traag, L. Waltman, and N. J. van Eck. From louvain to leiden: guaranteeing well-connected communities. *Scientific Reports*, 9(1):5233, 2019. doi: 10.1038/s41598-019-41695-z. URL https://doi.org/10.1038/s41598-019-41695-z.

Quan M. Tran, Suong N. Hoang, Lam M. Nguyen, Dzung Phan, and Hoang Thanh Lam. TabularFM: An open framework for tabular foundational models. *arXiv preprint arXiv:2406.09837*, 2024.

Yazheng Yang, Yuqi Wang, Guang Liu, Ledell Wu, and Qi Liu. UniTabE: A universal pretraining protocol for tabular foundation model in data science. In *International Conference on Learning Representations (ICLR)*, 2024.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, pp. 818–833. Springer, 2014.

Xiyuan Zhang, Danielle C. Maddix, Junming Yin, Nick Erickson, Abdul Fatir Ansari, Boran Han, Shuai Zhang, Leman Akoglu, Christos Faloutsos, Michael W. Mahoney, Cuixiong Hu, Huzefa Rangwala, George Karypis, and Bernie Wang. Mitra: Mixed synthetic priors for enhancing tabular foundation models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Bingzhao Zhu, Xingjian Shi, Nick Erickson, Mu Li, George Karypis, and Mahsa Shoaran. XTab: Cross-table pretraining for tabular transformers. In *International Conference on Machine Learning (ICML)*, 2023.

A Tabular Foundation Models

We study tabular foundation models (TFMs) that span three major design paradigms (transformer ICL, hypernetworks, and finetuned language models), selected for architectural diversity and public availability. The results section focuses on the six transformer-based models; HyperFast and Tabula-8B are included in geometric comparisons but excluded from causal analysis due to architectural constraints on intervention design.

B Data Preprocessing

We follow the AutoGluon (Erickson et al., 2020) preprocessing pipeline used in the TabArena benchmark (Erickson et al., 2025). For each dataset, preprocessing is applied identically across all models so that differences in embedding geometry reflect model architecture rather than input representation differences.

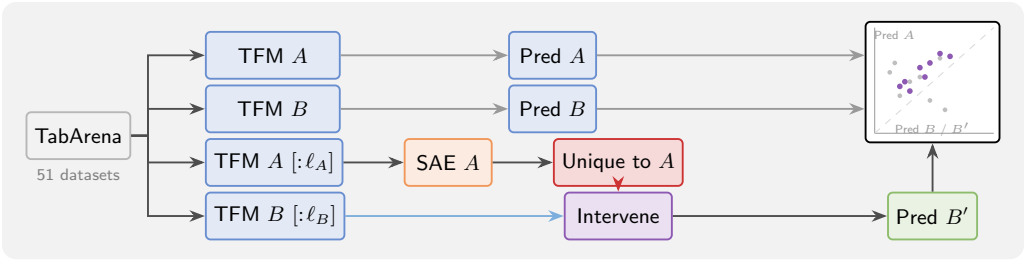


Figure 4: Concept intervention pipeline. Test rows from TabArena pass through two TFMs: baselines (TFM A, TFM B) produce reference predictions; the intervention path extracts the sparse-autoencoder concepts unique to the stronger model (TFM A) and injects them into the weaker model (TFM B) at the embedding layer, yielding improved predictions B'. The scatter plot shows B' predictions moving closer to A (purple) compared to the B baseline (grey).

Model	Category
TabPFN v2.5 (Hollmann et al., 2025)	Synthetic-data pretrained
Mitra (Zhang et al., 2025)	Synthetic-data pretrained
TabICL v2 (Qu et al., 2025)	Synthetic-data pretrained
TabICL (Qu et al., 2025)	Synthetic-data pretrained
TabDPT (Ma et al., 2025)	Real-data pretrained
CARTE (Kim et al., 2024)	Real-data pretrained
HyperFast (Bonet et al., 2024)	Hypernetwork
Tabula-8B (Gardner et al., 2024)	LLM-based

Table 4: Models evaluated, grouped by training paradigm.

Feature type inference. Categorical columns are identified by OpenML metadata where available; otherwise inferred from dtype and cardinality (integer columns with fewer than 20 unique values are treated as categorical). Remaining columns are treated as numeric.

Missing value imputation. Numeric missing values are imputed with the column median; categorical missing values receive a dedicated `NaN` category token.

Categorical encoding. Categorical features are ordinally encoded (integer codes) before passing to all models. Models with their own internal embedding layers (CARTE, Tabula-8B) perform further encoding internally; for these models the ordinal codes serve as indices.

Numeric scaling. No global scaling is applied at the preprocessing stage—individual models apply their own internal standardization. SAE training uses a separate per-dataset StandardScaler fit on the training split of each fold (see Appendix D).

Train/test splits. We use the TabArena standardized outer fold splits. For embedding extraction, two folds are used as the context (training) set and the remaining fold as the query (test) set. The same test-fold embeddings are used for SAE training data construction, concept labeling, and ablation evaluation to ensure consistency throughout the pipeline.

C Representation Extraction Layer Selection

Transformer-based models that undergo pretraining develop layered representations where early layers learn general concepts and later layers specialize for the pretraining objective. Extracting from an inappropriate layer would yield a representation optimized for the training loss rather than the general-purpose embedding geometry we wish to study. We therefore conduct a systematic layer selection analysis for each model using Centered Kernel Alignment (CKA) between the input layer

(L0) and every subsequent layer, computed across all 51 TabArena datasets (38 classification, 13 regression) for models supporting both task types, or across the 38 classification datasets for the classification-only models (TabICL v1, HyperFast).

We define the *critical layer* as the first layer where CKA with the input drops below 0.5 (i.e., the representation has lost more than half its input structure). Our default extraction point is the rounded mean critical layer per architecture variant, where "variant" groups datasets by (`n_layers`, task type) so that, e.g., the TabPFN classifier checkpoint (24 layers) and regressor checkpoint (18 layers) are selected independently. A small number of models (CARTE, HyperFast, Tabula-8B) deviate from the default rule for architectural reasons noted in their per-model subsections. The 0.5 threshold follows the change-point convention used by prior CKA analyses of layered representations (Kornblith et al., 2019; Liu et al., 2025).

Model	Architecture	Layers	Extract	Depth	n	Notes
TabPFN v2	Transformer (ICL)	24	L19	79%	38	Classifier checkpoint
TabPFN v2	Transformer (ICL)	18	L11	61%	13	Regressor checkpoint
Mitra	2D Attention	13	L12	92%	51	cls/reg converge to final-norm
TabICL	Column-then-Row TFM	13	L10	77%	38	v1 (classification only)
TabICL-v2	Column-then-Row TFM	13	L12	92%	51	cls/reg converge
TabDPT	Transformer + Retrieval	17	L14	82%	51	cls/reg converge
CARTE	Graph Transformer	5	L1	20%	51	cls/reg converge
HyperFast	Hypernetwork	3	L2	67%	38	Classification only; excluded from causal
Tabula-8B	Llama-3 8B (LLM)	33	L13	39%	51	44 at L13, 7 at L10; excluded from causal

Table 5: Embedding extraction points per model, selected by layerwise CKA critical depth analysis across TabArena datasets. **Layers** = total extraction points (including input); **Extract** = optimal layer (round-to-nearest of the per-variant mean critical layer; CARTE deviates as noted in §C.7); **Depth** = extraction layer / total layers; n = datasets analyzed. Layers are selected per architecture variant grouped by (`n_layers`, task type); rows above the midrule are the seven models retained for causal analysis (§4.2). TabPFN uses different checkpoints for classification (24 layers) and regression (18 layers), so the two task variants are listed separately. Mitra, TabICL-v2, TabDPT, and CARTE perform independent cls/reg selection but converge to the same layer; their $n = 51$ row reflects the merged sweep. Source: `output/sae_training_round10/<model>_taskaware_norm_stats.npz`.

C.1 Threshold sensitivity

To verify that the 0.5 cutoff is not a load-bearing tuning choice, we replay the per-dataset L0–CKA traces through the same selection rule at thresholds $T \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$ and report how the chosen critical layer shifts. Table 6 reports the layer shift relative to $T = 0.5$. Because the extraction layer is computed from the per-variant mean critical layer (round-to-nearest), threshold-induced shifts in the critical layer translate directly to corresponding shifts in the extraction layer.

The wide threshold-stability of the four cliff-shaped models (Mitra cls/reg, TabICL-v2, CARTE) reflects the structure of their drift profiles in Figure 5: any threshold in $[0.3, 0.5]$ lands on the same post-cliff layer. The two models that would shift substantially under a different threshold (Tabula-8B, with no sharp cliff; and the bimodal TabPFN regressor) are either already excluded from the causal analyses or have their profile irregularity flagged in the per-model subsections below. Within the conservative ± 1-layer band, we expect adjacent extraction layers to encode largely overlapping concepts, since the embedding geometry varies smoothly across nearby layers and probing studies of similar tabular foundation models report stable feature semantics in such windows (Gupta et al., 2026); we therefore do not retrain the SAE dictionaries at the neighboring layers.

C.2 Combined depth profile

Figure 5 overlays the mean CKA drift profiles for all seven models on a normalized depth axis ($0 =$ input, $1 =$ final layer). The four transformer-based models (TabPFN, TabICL, Mitra, TabDPT) share a common pattern: CKA remains above 0.8 through roughly 60% of depth, then drops sharply.

Model	T=0.3	T=0.4	T=0.5	T=0.6	T=0.7
Mitra (cls)	+0	+0	+0	+0	−1
Mitra (reg)	+0	+0	+0	+0	−2
TabICL-v2	+0	+0	+0	−1	−3
CARTE	+1	+1	+0	+0	+0
TabDPT	+2	+1	+0	−1	−2
TabPFN (cls)	+2	+1	+0	−1	−3
TabPFN (reg)	+3	+2	+0	−2	−4
TabICL	+2	+1	+0	−2	−6
HyperFast	+0	+0	+0	+0	+0
Tabula-8B	+9	+4	+0	−1	−3

Table 6: Critical-layer shift relative to the paper's $T = 0.5$ choice when the threshold is varied. Models above the rule are included in the causal analyses (§4.2); HyperFast and Tabula-8B (below the rule) are excluded. For all six causal models the shift is at most 1 layer in $T \in [0.4, 0.6]$, and four (Mitra cls/reg, TabICL-v2, CARTE) are unchanged across the wider range $T \in [0.3, 0.5]$. TabICL is the most threshold-sensitive causal model, consistent with its smoother drift profile (§C.5). Generated by `scripts/analysis/cka_threshold_sensitivity.py` from the cached per-dataset CKA profiles.

CARTE and HyperFast, with only 3–5 layers, show immediate transformation. Tabula-8B drifts gradually with no sharp cliff, consistent with its LLM architecture where representations change smoothly across 33 layers.

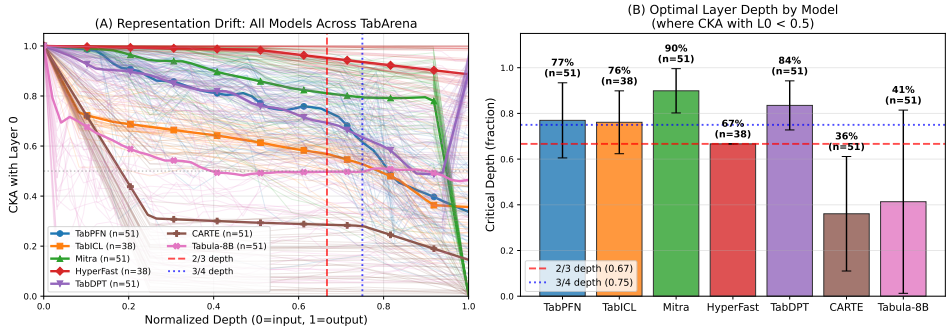


Figure 5: Representation drift from input (CKA with L0) as a function of normalized depth across all seven models. Transformer models share a common drift profile with a sharp cliff at 70–90% depth, while non-transformer architectures (CARTE, HyperFast) and the LLM (Tabula-8B) follow distinct patterns.

C.3 TabPFN

TabPFN's classifier is a 24-layer in-context learning transformer pretrained on synthetic classification data via structural causal models. Its CKA profile (Figure 6) shows gradual drift through the first 14 layers, then a steep transition around L18–19 where the representation commits to the classification objective. Across 38 classification datasets the mean critical layer is L19.2 ($\sigma = 2.5$); we extract at L19 (mean CKA ≈ 0.45 with input, 79% depth).

TabPFN also provides a separate regressor checkpoint with 18 layers (vs. 24 for the classifier). The regressor critical depth is bimodal: four datasets (concrete, Food_Delivery, miami_housing, superconductivity) have early critical layers at L5–6 (∼30% depth), while the remaining nine cluster at L14–17 (∼85%). Across all 13 regression datasets the mean critical layer is L12.2 ($\sigma = 4.5$); we use a fixed extraction layer L11 (mean CKA ≈ 0.66 with input, 61% depth) for all TabPFN regression tasks.

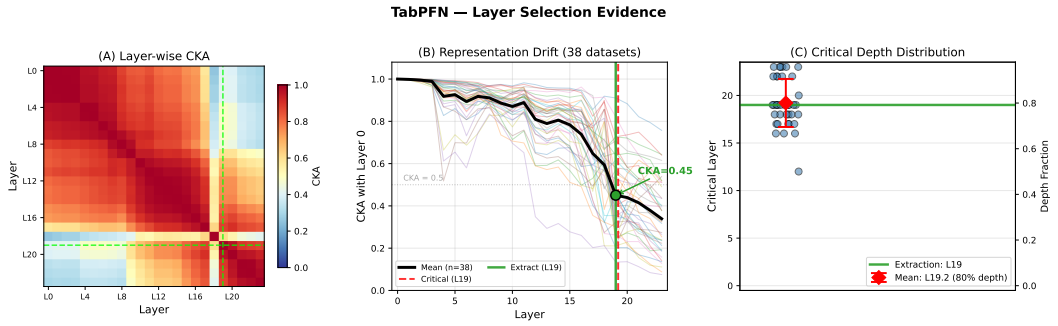


Figure 6: TabPFN layer selection evidence. **(A)** CKA heatmap for a representative dataset (adult), with extraction layer (green) and critical layer (red) marked. **(B)** CKA drift from input across 38 classification datasets, showing the late-stage transition into the classification head. **(C)** Critical depth distribution; mean critical layer is L19.2 (80% depth).

C.4 MITRA

Mitra is a 72M-parameter transformer with 2D attention, accessed through AutoGluon's sklearn interface. Its 13 extraction points correspond to 12 transformer layers plus a final normalization layer. Mitra uses *separate pretrained checkpoints* for classification (`autogluon/mitra-classifier`) and regression (`autogluon/mitra-regressor`), producing geometrically unrelated embeddings (per-dimension correlation $r \approx 0.02$). We therefore conduct independent layer selection for each checkpoint, and both checkpoints converge on L12 – the final normalization layer.

Classifier. The classifier (Figure 7) maintains stable CKA > 0.9 through L8, then drops sharply, with mean critical layer L11.6 ($\sigma = 1.4$) across 38 classification datasets. The final normalization layer (L12) drives within-input CKA toward zero (mean CKA ≈ 0.004 with L0), yet is also the layer that maximizes *cross-model* CKA against other transformers' deep extractions: within-model and cross-model CKA target different geometries here, and the round-to-nearest selection rule, together with our preference for cross-model alignment, places extraction at L12.

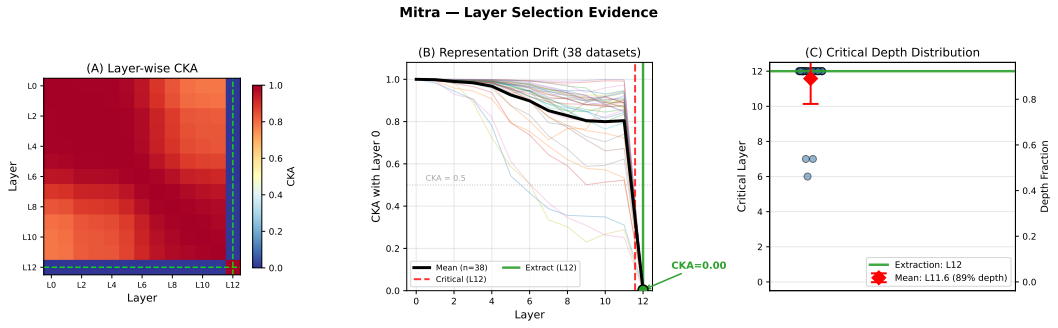


Figure 7: Mitra classifier layer selection evidence across 38 TabArena classification datasets. Mean critical layer is L11.6 (89% depth).

Regressor. The regressor (Figure 8) shows a striking result: all 13 regression datasets have identical critical depth at L12 (92% depth) with zero variance. This uniformity contrasts with the classifier, where a handful of datasets deviate. We extract at L12, matching the classifier; the final normalization layer again drives within-input CKA toward zero, confirming this is an architectural property of Mitra rather than a checkpoint-specific artifact.



Figure 8: Mitra regressor layer selection evidence across 13 TabArena regression datasets. All datasets converge to critical depth at L12 (92% depth) with zero variance.

C.5 TabICL

TabICL (v1) uses a column-then-row attention architecture with 13 extraction points and supports classification only. Its CKA profile (Figure 9) shows smooth, progressive drift without the sharp cliff characteristic of other transformers—the column-then-row attention pattern produces more gradual representation changes. Across 38 classification datasets the mean critical layer is L9.9 ($\sigma = 1.8$); we extract at L10 (mean CKA ≈ 0.47 with input, 77% depth). TabICL-v2, which supports both classification and regression and which we use elsewhere in the paper, is selected independently and converges to L12 for both task types.

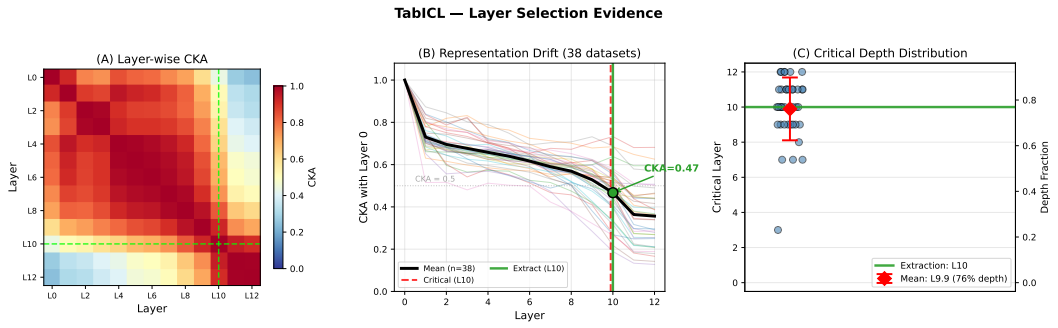


Figure 9: TabICL layer selection evidence across 38 TabArena classification datasets. Mean critical layer is L9.9 (76% depth). The column-then-row attention architecture produces smoother drift than standard transformers.

C.6 TabDPT

TabDPT combines a transformer with retrieval-augmented attention across 17 extraction points. Its drift profile (Figure 10) shows gradual decay through L10, followed by rapid divergence. The retrieval mechanism appears to stabilize early-layer representations while concentrating task-specific adaptation in the final layers. Across 51 datasets the mean critical layer is L14.2 ($\sigma = 1.8$); we extract at L14 (mean CKA ≈ 0.49 with input, 82% depth). The classifier and regressor variants of TabDPT share the same checkpoint and converge to the same extraction layer.

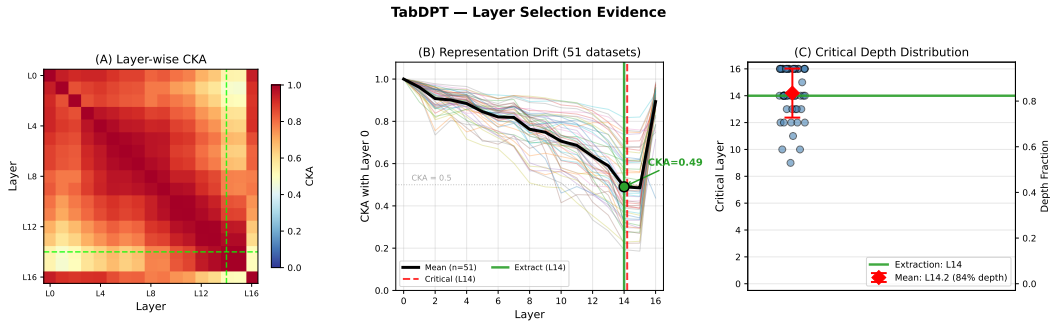


Figure 10: TabDPT layer selection evidence across 51 TabArena datasets. Mean critical layer is L14.2 (84% depth).

C.7 CARTE

CARTE is a graph neural network that constructs a star graph from tabular concepts, with 5 extraction points. Its CKA profile (Figure 11) reveals a dramatic transformation at the first GNN layer: CKA drops from 1.0 to a mean of ≈ 0.31 between L0 and L1, as the star graph construction fundamentally reshapes the representation. Subsequent layers continue to diverge. Across 51 datasets the mean critical layer is L1.8 ($\sigma = 1.2$); we override the round-to-nearest rule (which would place extraction at L2) and extract at L1 (20% depth) for both classification and regression. This is the earliest extraction point among all models, reflecting that GNN message passing produces immediate, large-scale structural changes rather than the gradual drift seen in transformers, and that the deeper CARTE layers specialize past the point of useful general-purpose embedding.

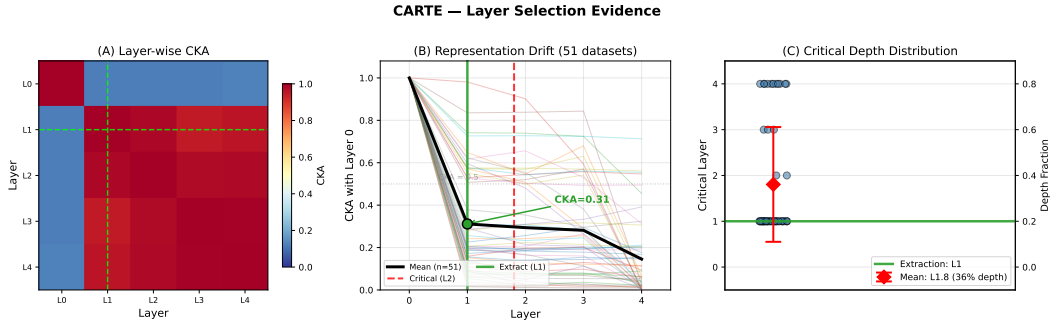


Figure 11: CARTE layer selection evidence across 51 TabArena datasets. The GNN star graph construction produces an immediate representation shift at L1. Mean critical layer is L1.8 (36% depth).

C.8 HyperFast

HyperFast is a hypernetwork that generates dataset-specific 3-layer MLPs (784→784→784→ n_{classes} with residual connections) and supports classification only. The generated network has only 3 extraction points: across 38 classification datasets the critical layer is always L2 ($\sigma = 0$, zero variance), so the round-to-nearest rule places extraction at L2 (the layer feeding the classification head, 67% depth, mean CKA ≈ 0.89 with input; Figure 12). HyperFast appears in the geometric diagnostics only; the near-identity transformation through L1 and the proximity of L2 to the classification head together make HyperFast an inappropriate input for the SAE-based concept analysis, so it is excluded from the §4.2 causal results.

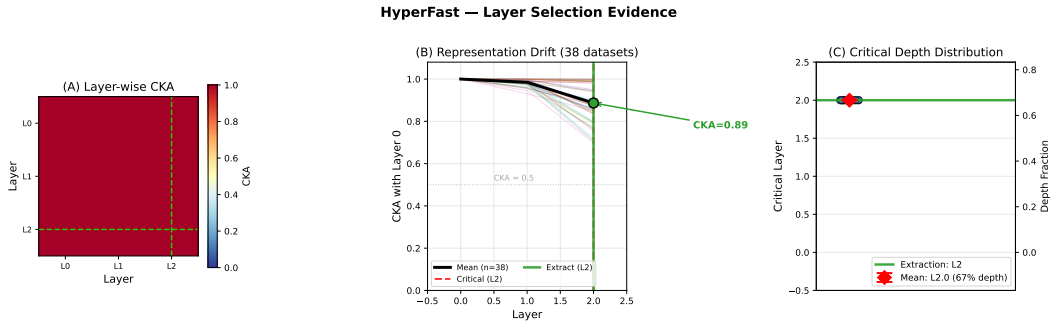


Figure 12: HyperFast layer selection evidence across 38 TabArena classification datasets. The generated MLP preserves input structure almost perfectly through L1 (CKA ≈ 1.00) and only diverges at the classification head (L2).

C.9 TABULA-8B

Tabula-8B is a Llama-3 8B model fine-tuned on serialized tabular data, with 33 extraction points (32 transformer layers + embedding). Unlike the smaller transformer models, its CKA profile (Figure 13) shows highly variable per-dataset drift with no sharp cliff. Across 51 datasets the mean critical layer is L13.7 ($\sigma = 13.2$); the task-aware grouping yields L13 for 44 datasets and L10 for 7 (the bimodality reflects the wide critical-layer dispersion rather than a clean cls/reg split, with 5 of the L10 datasets being classification and 2 regression). Tabula-8B appears in the geometric diagnostics only; its inference is bounded to 32 context rows, which makes most imbalanced classification tasks infeasible, so it is excluded from the §4.2 causal results.

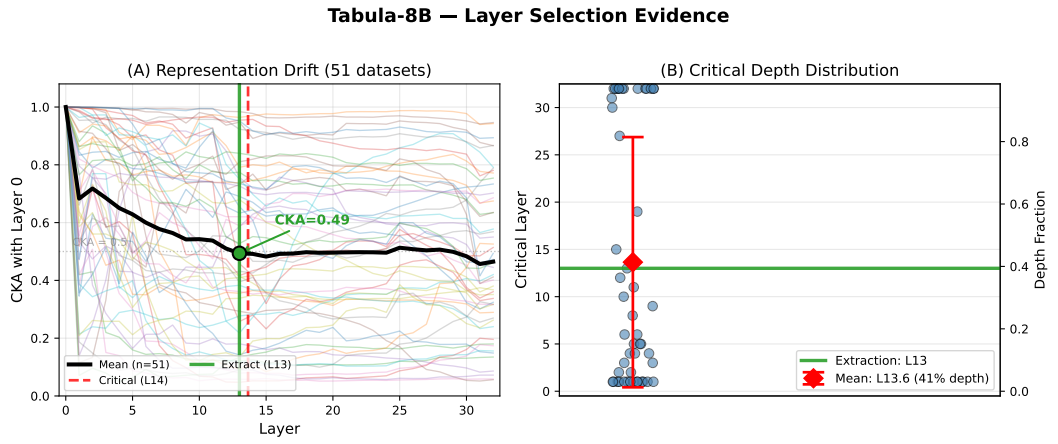


Figure 13: Tabula-8B layer selection evidence across 51 TabArena datasets. Per-dataset CKA drift with very high variance and no sharp cliff, unlike the transformer-only models.

C.10 CKA FOR REGRESSION MODELS

C.11 PROCRUSTES RESIDUALS

As a confirming geometric check, we solve the optimal orthogonal alignment between each pair of model embeddings (Schönemann, 1966):

$$W^* = \arg\min_{W^\top W = I} \|XW - Y\|_F^2. \tag{3}$$

The residual $\|XW^* - Y\|_F$ after alignment indicates how much of the embedding difference is explained by rotation versus more complex transformations.

Table 7: Linear CKA similarity ($\in [0, 1]$) on 13 regression TabArena datasets. TabPFN uses layer 11 for regression (vs. layer 19 for classification) due to differing regressor architecture (18 vs. 24 layers). Regression CKA values are consistently higher than classification (Figure 2), suggesting stronger representational convergence on continuous targets.

	TabPFN	Mitra	TabICL-v2	TabDPT	CARTE
TabPFN	1	–	–	–	–
Mitra	0.76±0.12	1	–	–	–
TabICL-v2	0.66±0.17	0.83±0.16	1	–	–
TabDPT	0.77±0.12	0.87±0.12	0.87±0.10	1	–
CARTE	0.48±0.15	0.56±0.21	0.47±0.23	0.53±0.21	1

Table 8: Procrustes disparity ($d^2 = 1 - \mathrm{tr}(R^\top A^\top B)$, $\in [0, 1]$) after optimal orthogonal alignment. 38 TabArena classification datasets, 3768 pooled samples (restricted to datasets shared by all 8 models).

	TabPFN	Mitra	TabICL	TabICL-v2	TabDPT	CARTE	HyperFast	Tabula-8B
TabPFN	0	–	–	–	–	–	–	–
Mitra	0.4737	0	–	–	–	–	–	–
TabICL	0.5519	0.4527	0	–	–	–	–	–
TabICL-v2	0.5192	0.3613	0.4388	0	–	–	–	–
TabDPT	0.5478	0.4517	0.3820	0.4414	0	–	–	–
CARTE	0.7547	0.7242	0.7605	0.7471	0.7584	0	–	–
HyperFast	0.8070	0.7785	0.7333	0.7495	0.7002	0.8604	0	–
Tabula-8B	0.7338	0.6902	0.6496	0.6783	0.6134	0.8011	0.6627	0

D Sparse Autoencoder Training

We train Matryoshka-Archetypal Sparse Autoencoders (SAEs) to decompose each model's embedding space into interpretable concept dictionaries. This appendix details the architecture selection, training procedure, and per-model results.

D.1 Data preparation

We make heavy use of the openml recorded splits from TabArena. For inference, as previously described, the first 2 outer folds are reserved for context, the last outer fold is used for query. We reuse that outer fold to build our SAE training set.

We do doubly stratified sampling on each dataset. For classification tasks we stratify on the target variable; for regression tasks we stratify on quartiles. The second layer of stratification is on loss using TabPFN predictions - where we sample on terciles, essentially breaking the dataset into 3 difficulties: easy, medium, and hard. We cap the number of samples per dataset to 700 (500 train, 200 test) in an attempt to give equal representation across the 51 datatsets. For small datasets where the number of samples in the fold is less than 700, we take all the samples. The table below details those small datasets where we have taken all the samples for SAE training.

D.2 Architecture-Specific Losses

We evaluate five SAE architectures before selecting the final variant. All use a single hidden layer with ReLU activations; they differ in how sparsity and decoder constraints are imposed.

L1 SAE.

$$\mathcal{L}_{\text{L1}} = \|x - W_d h\|_2^2 + \lambda \|h\|_1 \tag{4}$$

Table 9: Datasets below the 700-row holdout threshold. These datasets contribute all available holdout rows to the SAE corpus rather than the standard 500/200 train/test sample. The train/test split maintains the same 71/29 ratio used for larger datasets.

Dataset	Task	Holdout	Train	Test	% of Cap
blood-transfusion-service-center	C	250	179	71	36%
diabetes	C	256	183	73	37%
anneal	C	300	214	86	43%
QSAR_fish_toxicity	R	303	216	87	43%
credit-g	C	334	239	95	48%
maternal_health_risk	C	338	241	97	48%
concrete_compressive_strength	R	344	246	98	49%
qsar-biodeg	C	352	251	101	50%
healthcare_insurance_expenses	R	446	319	127	64%
website_phishing	C	451	322	129	64%
Fitness_Club	C	500	357	143	71%
airfoil_self_noise	R	501	358	143	72%
Another-Dataset-on-used-Fiat-500	R	513	366	147	73%
MIC	C	567	405	162	81%
Is-this-a-good-customer	C	575	411	164	82%
Total (15 datasets)		**6030**	**4307**	**1723**	

TopK SAE.

$$\mathcal{L}_{\text{TopK}} = \|x - W_d \,\text{TopK}(h, k)\|_2^2 \tag{5}$$

where $\text{TopK}(h, k)$ retains the k largest activations and zeros the rest, providing hard sparsity without a tunable λ.

Matryoshka SAE (Bussmann et al., 2025).

$$\mathcal{L}_{\text{Mat}} = \sum_{s \in \mathcal{S}} w_s \|x - W_d^{(1:s)} h_{1:s}\|_2^2 \tag{6}$$

where $\mathcal{S} = \{s_1, s_2, \ldots, s_m\}$ are nested concept subsets and $h_{1:s}$ are the first s latent dimensions. The loss forces early concepts to be broadly useful, while later concepts specialize.

Archetypal SAE (Fel et al., 2025).

$$\mathcal{L}_{\text{Arch}} = \|x - \hat{x}\|_2^2 + \lambda \|h\|_1, \qquad W_d = C \cdot \text{softmax}(\phi/\tau) \tag{7}$$

where $C \in \mathbb{R}^{d \times K}$ are K-means centroids of the training data (one centroid per column), $\phi \in \mathbb{R}^{K \times n_{\text{dict}}}$ are learnable archetypal logits with softmax taken along the K axis so each column gives convex weights summing to 1, and τ is a temperature parameter. Each decoder atom is a convex combination of centroids, constraining the dictionary to the convex hull of the data and providing geometric stability.

Matryoshka-Archetypal SAE.

$$\mathcal{L}_{\text{Mat-Arch}} = \sum_{s \in \mathcal{S}} w_s \|x - W_d^{(s)} \, h_{1:s}\|_2^2, \qquad W_d^{(s)} = C \cdot \text{softmax}(\phi^{(s)}/\tau) \tag{8}$$

Combines the nested multi-scale loss of Matryoshka with the archetypal decoder constraint. This produces a hierarchical dictionary where concepts are ordered by abstraction level (coarse $\rightarrow$ fine) and decoder atoms are anchored to data geometry.

D.3 Architecture Selection

We conduct an architecture comparison on TabPFN embeddings (21 TabArena test-set datasets) using Optuna (Akiba et al., 2019) sweeps of 30 trials per architecture. Table 10 summarizes the results. The composite score $= 0.4 \times (1 - \text{RMSE}) + 0.6 \times \text{Stability}$ emphasizes reproducibility,

Table 10: SAE architecture comparison on TabPFN embeddings (21 TabArena test-set datasets). All architectures use TopK sparsity and were optimized via Optuna (30 trials each).

Type	Hyperparameters	Score	RMSE	Stab	L_0	Dead%
Mat-Archetypal	m=8, k=128, n=256, τ=0.09	**0.928**	0.620	**0.917**	105	**13.2**
Archetypal	m=8, k=32, n=256, τ=0.15	**0.928**	**0.528**	0.914	32	58.2
TopK	m=8, k=16	0.714	0.436	0.543	16	83.1
Matryoshka	m=4, k=32	0.593	0.293	0.336	176	66.7

Hyperparameters: m = expansion factor, k = TopK, n = archetypes, τ = simplex temperature.
Metrics: Stab = s_n^{dec} decoder stability, L_0 = avg. active concepts, Dead% = fraction of never-activated dictionary atoms.

since concept comparison across models requires that learned concepts are stable across independent training runs.

The Matryoshka-Archetypal architecture ties for the best composite score (0.928) while achieving the lowest dead neuron rate (13.2%). The archetypal decoder constraint provides 2.7× better stability than Matryoshka alone (0.92 vs 0.34). Figure 14 illustrates the stability–reconstruction tradeoff across architectures and models: Archetypal SAEs consistently achieve >0.6 stability where standard and TopK SAEs plateau below 0.4.

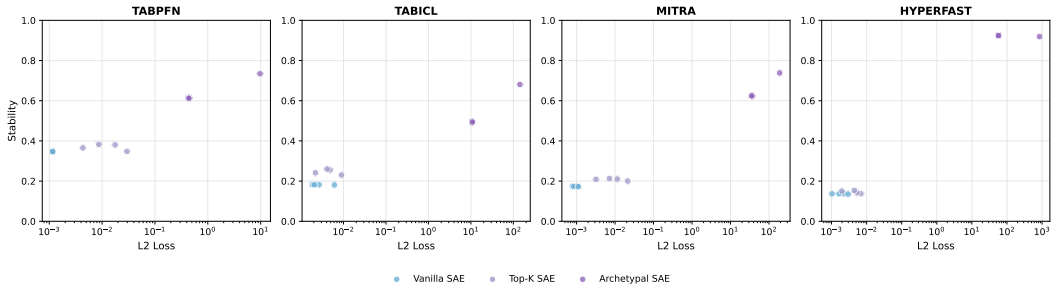


Figure 14: Stability vs. reconstruction loss across three SAE architectures on four models. Archetypal SAEs (dark purple) trade modestly higher loss for dramatically better decoder stability. Vanilla and TopK SAEs cluster below 0.4 stability regardless of loss.

D.4 Stability Metric

We measure cross-run reproducibility using the s_n^{dec} metric from Fel et al. (2025), which computes the mean cosine distance between decoder weight vectors across independent training runs:

$$s_n^{\text{dec}} = \frac{1}{n_{\text{dict}}} \sum_{i=1}^{n_{\text{dict}}} \max_j \frac{W_d^{(i)} \cdot W_d^{(j),\text{ref}}}{\|W_d^{(i)}\|_2 \, \|W_d^{(j),\text{ref}}\|_2} \tag{9}$$

where the max is over the best-matching concept in the reference run (Hungarian matching). Values closer to 1.0 indicate that the same dictionary atoms are learned regardless of random initialization. For our purposes, stability > 0.7 indicates the dictionary is reproducible enough for cross-model concept comparison.

D.5 Dead Neuron Revival

Large SAE dictionaries suffer from concept death, where TopK selection permanently kills marginal concepts. In Archetypal SAEs, near-duplicate decoder atoms (cosine similarity > 0.9 between convex hull neighbors) make concepts interchangeable, amplifying this effect.

For all production Matryoshka-Archetypal SAEs we use AuxK (Gao et al., 2025) with $\alpha = 1/32$ together with periodic dead-neuron resampling (every 25,000 steps, 1024 reactivation samples) as

the revival mechanism; the multi-scale Matryoshka loss further encourages concepts at each scale band to contribute to reconstruction. With these mechanisms in place, the production SAEs achieve 81.5–98.1% alive concepts across the eight models (Table 11), and the Matryoshka-Archetypal entry of the architecture-comparison sweep on TabPFN shows 13.2% dead concepts at $8\times$ expansion (Table 10).

D.6 Hyperparameter Optimization

For each model we run an Optuna (Akiba et al., 2019) hyperparameter sweep using a TPE sampler with 30 trials per model. Each trial fits an SAE for 100 epochs on the training corpus and is scored on a held-out split of the same corpus that is never seen by the training loop. The search space covers the dictionary expansion factor (chosen from $\{4, 8, 16\}$ for $d_{\text{model}} < 2048$ and $\{1, 2, 4\}$ otherwise, to cap SAE parameter counts), top-k sparsity ($k \in \{16, 32, 64, 128, 256\}$), learning rate ($\log \mathcal{U}[5\text{e}{-5}, 2\text{e}{-3}]$), and the archetypal constraints — temperature ($\log \mathcal{U}[0.05, 0.5]$), relaxation ($\mathcal{U}[0, 2]$), and number of archetypes (up to 2000). Dead-neuron mitigation (AuxK (Gao et al., 2025) with $\alpha = 1/32$ and periodic resampling) is held fixed at literature defaults rather than searched, to prevent the sampler from exploiting aux-loss shrinkage as a cheap way to lower the objective.

Objective. The Optuna objective combines reconstruction, capacity, and sparsity into a single scalar:

$$\mathcal{J} = \ell^{\text{test}}_{\text{recon}} \cdot \sqrt{d_{\text{hidden}}} \cdot \sqrt{L_0} \,/\, \rho_{\text{alive}},$$

where $\ell^{\text{test}}_{\text{recon}}$ is held-out MSE, d_{hidden} is the SAE dictionary size, L_0 is the average number of active latents per input, and ρ_{alive} is the fraction of latents that fire on at least one batch. The $\sqrt{d_{\text{hidden}}}$ and $\sqrt{L_0}$ factors push the sampler toward smaller, sparser dictionaries when reconstruction is comparable; dividing by ρ_{alive} penalizes configurations that waste capacity on dead features. After the sweep, we inspect the top candidates on a Pareto plot of L_0 against reconstruction R^2, alive fraction, and cross-seed stability (correlation of decoder columns between independently trained seeds), selecting a configuration that clears $R^2 \geq 0.80$, alive fraction ≥ 0.80, and stability ≥ 0.75.

Data preparation. Embeddings from all TabArena datasets are pooled into a single training set per model. Each embedding is mean-centered; an internal BatchNorm layer handles per-dimension variance normalization, ensuring consistent scaling between training and evaluation.

Mitra special case. The Mitra classifier and regressor use separate pretrained checkpoints producing geometrically unrelated embeddings ($r \approx 0.02$ per-dimension correlation; Section C.4). We train independent SAEs for classification (38 datasets) and regression (13 datasets).

D.7 Per-Model Results

Table 11 summarizes the validated Matryoshka-Archetypal SAE for each model. All SAEs use the same five-band Matryoshka structure ($d_{\text{SAE}}/16,\ /8,\ /4,\ /2,\ /1$), enabling cross-model concept comparison at matched relative abstraction levels.

D.8 Matryoshka Band Variance

Figure 15 reports cumulative reconstruction variance by Matryoshka band for each model's trained SAE. Concepts are sorted by contribution within each band. Band b0 captures 66–84% of variance; bands b0+b1 reach 84–95%, confirming that the nested-loss objective concentrates reconstruction in the coarse bands as designed.

E SAE Concept Analysis

We interpret each model's SAE dictionary by measuring how individual concepts respond to computable row-level meta-concepts. This appendix describes the labeling methodology, presents per-model coverage results, and analyzes the hierarchical structure of the Matryoshka scale bands.

Model	d	d_{SAE}	TopK	LR	K	τ	Recon	%Alive	s_n^{dec}
TabPFN	192	768	256	2.5e-04	1000	0.20	0.0527	81.5	0.777
CARTE	300	1200	256	2.5e-04	2000	0.48	0.0233	90.2	0.809
TabICL	512	2048	256	8.4e-05	512	0.26	0.1925	90.2	0.839
TabICL-v2	512	2048	256	8.4e-05	512	0.07	0.2595	98.1	0.939
TabDPT	768	3072	128	3.4e-04	1000	0.22	0.2722	81.8	0.906
Mitra	512	2048	32	1.7e-04	1000	0.06	0.1543	83.5	0.942
HyperFast	784	3136	16	6.2e-04	1000	0.07	0.6189	83.0	0.917
Tabula-8B	4096	4096	64	1.1e-04	512	0.11	0.3516	88.4	0.907

Table 11: SAE configuration and quality metrics per model (Matryoshka-Archetypal architecture). d = embedding dim, d_{SAE} = dictionary size, K = archetypes, τ = simplex temperature, Recon = MSE reconstruction loss, %Alive = percentage of active dictionary elements, s_n^{dec} = decoder stability across seeds. Sparsity is enforced structurally by TopK, so no L1 penalty is applied to the loss (§D.5).

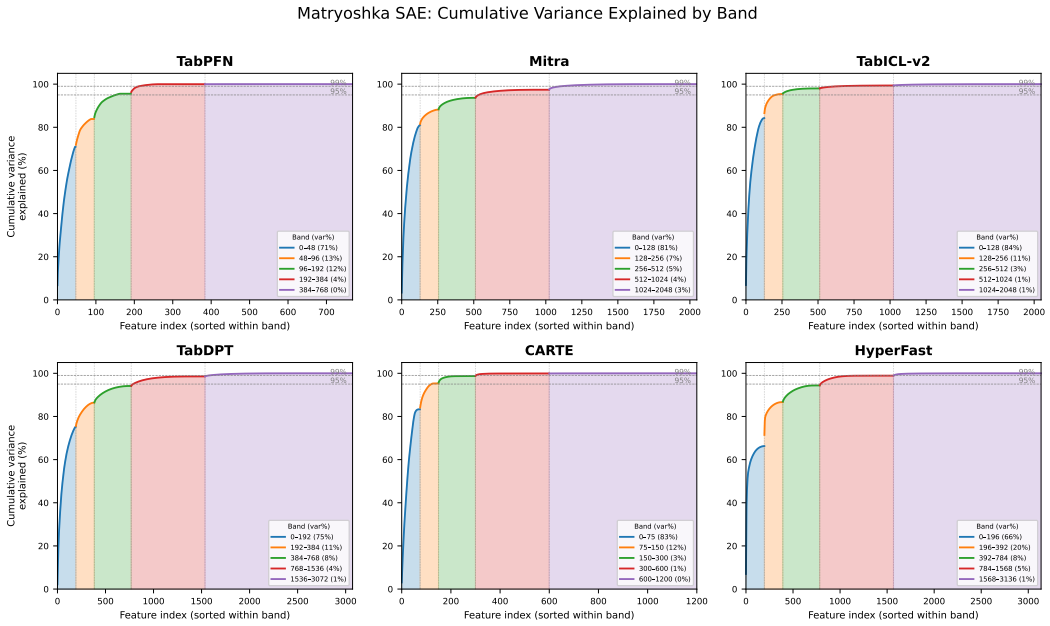


Figure 15: Cumulative variance explained by Matryoshka band for each model's trained SAE. Concepts sorted by contribution within each band. Band b0 captures 66–84% of variance; bands b0+b1 reach 84–95%.

Dense latents. A subset of each dictionary's concepts are *dense*, firing on a large fraction of inputs. Recent work argues that such dense latents are genuine features rather than training artifacts (Sun et al., 2025), so their presence is not a defect in our dictionaries. Density does, however, make these concepts harder to label: a latent that fires almost everywhere leaves little discriminative activating-versus-non-activating evidence to pin down what it encodes, which is a concrete source of the labeling difficulty for part of each dictionary.

E.1 Concept Matching

To reduce the dimensionality of the task, we first compute pairwise matching of concepts between SAEs. The first step is to use a greedy mutual nearest neighbor (MNN) approach to find pairs. This methodology is first applied between trained and randomly initialized SAEs to test the power of the approach.

E.1.1 Pairwise comparison of matched concepts

After MNN matching with co-occurrence pruning and Leiden community detection, the matching pipeline produces concept groups; each group contains the features from one or more models that detect a common pattern. Table 12 reports the number of groups containing features from each pair of models.

	TabPFN	Mitra	TabDPT	TabICL	TabICL-v2	CARTE
TabPFN	—	123	134	94	116	80
Mitra	123	—	467	189	323	219
TabDPT	134	467	—	229	374	240
TabICL	94	189	229	—	308	123
TabICL-v2	116	323	374	308	—	190
CARTE	80	219	240	123	190	—

Table 12: Pairwise shared-concept counts across the six models in §4.2 (HyperFast and Tabula-8B excluded). Each cell shows the number of MNN concept groups containing at least one feature from both the row and column model. Totals: 846 multi-model groups; 45 groups present in all six models.

E.1.2 Matching in the face of random initialization

The outlier in this group is CARTE - with the highest and most confident match rate. This is largely driven by the fact that the optimal layer found for CARTE was layer 1, where the embeddings have hardly been transformed beyond their original input geometry - and have not had a chance to transform into richer representations.

Results from MNN matching on trained models:

We use a noise threshold obtained from looking at matches between a trained SAE A vs a random SAE B. Matches between trained SAEs A and B need to exceed the threshold (p-90 value). This induces a bi-directional match over the 28 pairs of SAEs.

E.2 Meta-concept Coverage Across Models

Figure 16 displays the maximum $|d|$ achieved by any alive concept at each Matryoshka scale band, for each model and all 37 meta-concepts. Several patterns are consistent across all models:

Universally well-covered concepts. Geometric concepts (centroid distance, nearest-neighbor distance), distributional concepts (fraction negative, skewness, kurtosis), dimensionality (n_{numeric}, n_{cols}, n_{rows}), and concept correlation are strongly represented in every model ($|d| > 2.0$), often at multiple scale bands simultaneously. These form the "core vocabulary" of tabular embeddings.

Universally absent concepts. All categorical concepts (rarity, modal fraction, entropy, unique/rare counts), all missing-value indicators, dataset sparsity, and target properties show $|d| \approx 0$ across all models and all scale bands. This is not a limitation of the SAE — it reflects that TabArena preprocessing converts categorical concepts to numeric encodings before embedding extraction, and that these models handle missingness internally before the embedding layer.

Band structure. Coverage is broadly distributed across bands rather than concentrated in b0 (coarsest). Geometric and distributional concepts appear at every band, suggesting the Matryoshka hierarchy encodes the same concept types at different levels of specificity rather than partitioning concepts by type.

E.3 Dictionary Completeness and Universality

Figure 17 quantifies two aspects of concept coverage. Panel A shows the number of meta-concepts covered ($|d| > 1.0$) by each model at each scale band. All four models achieve similar overall cover-

Table 13: Random baseline control: MNN matches between each model's trained SAE and a randomly initialized SAE with identical architecture. Match % is relative to alive concepts in the trained SAE. High false-match rates (CARTE, HyperFast) indicate embedding geometry dominates over learned SAE structure.

Model	Matched	Match %	Mean r
TabPFN	24	3.2%	0.503
TabICL	110	21.9%	0.402
TabDPT	149	5.4%	0.258
Mitra	191	10.5%	0.390
Tabula-8B	338	8.5%	0.285
CARTE	390	19.2%	0.702
HyperFast	458	8.8%	0.531
TabICL-v2	758	9.7%	0.191

Table 14: concept matching rates across seeds. "cross" = matched in ANY cross-model pair (union), "rand" = matched vs random baseline. A concept is counted as matched if it appears in at least one MNN pair.

Model	b0 cross	b0 rand	b1 cross	b1 rand	b2 cross	b2 rand	b3 cross	b3 rand	b4 cross	b4 rand	Total
TabPFN	10.4%	4.2%	14.6%	4.2%	25.5%	4.2%	48.6%	3.3%	76.6%	2.7%	54.0%
Mitra	8.6%	18.0%	44.0%	13.3%	51.7%	11.9%	65.6%	11.8%	72.4%	7.7%	60.0%
TabDPT	13.0%	3.6%	40.5%	9.5%	59.7%	12.2%	60.7%	4.7%	70.2%	3.4%	60.1%
TabICL	23.2%	8.0%	28.1%	18.0%	57.3%	20.8%	81.0%	30.0%	89.1%	37.0%	54.2%
TabICL-v2	30.3%	31.1%	45.0%	20.2%	42.3%	14.7%	45.2%	9.1%	48.6%	4.4%	45.3%
CARTE	6.0%	12.7%	18.0%	13.3%	28.8%	18.2%	46.8%	23.9%	55.6%	18.9%	42.3%
HyperFast	21.2%	12.5%	27.2%	11.8%	36.7%	11.5%	44.6%	8.7%	53.1%	6.7%	44.1%
Tabula-8B	23.0%	26.6%	31.8%	22.7%	27.8%	10.7%	36.5%	6.3%	54.3%	4.7%	42.6%

age (19–22 of 37 meta-concepts), with TabICL and TabDPT showing the most scale-dependent variation. Panel B shows how many models share each covered concept: 15 of 20 covered meta-concepts are detected by all four models, confirming that concept coverage is largely model-independent.

The 17 consistently uncovered meta-concepts (bottom of Figure 16) are the same across all models: the 8 categorical concepts, 3 missing-value indicators, dataset sparsity, target z-score, and several fine-grained distributional measures (mean z-score, std, IQR, positive/negative outlier fractions). The shared coverage pattern suggests these gaps reflect properties of the evaluation protocol (numeric-only meta-concepts after preprocessing) rather than model-specific blind spots.

E.4 Mitra concepts for Figure 1

F Causal Intervention Details

F.1 Pipeline Overview

F.2 Per-Row Concept Importance

All causal interventions operate on the SAE test split (up to 200 rows per dataset, held out from SAE training). For each model and dataset, we compute the causal importance of every SAE concept on every test row via single-concept ablation. Given a row's embedding, we encode it through the SAE, zero out a single concept's activation, and decode to obtain the ablated reconstruction. The delta between the ablated and original reconstruction is added back to the model's cached hidden state at the extraction layer, and the model's remaining layers produce a modified prediction. The resulting change in prediction loss is that concept's causal contribution to the row, and the full sweep yields a per-row, per-concept importance matrix used by both the ablation and transfer experiments below.

Figure 18 reports the row-averaged per-concept $|\Delta\text{pred}|$ decay, ranked descending. Unmatched concepts carry longer importance tails than matched concepts (panels (a) vs. (b)), particularly for

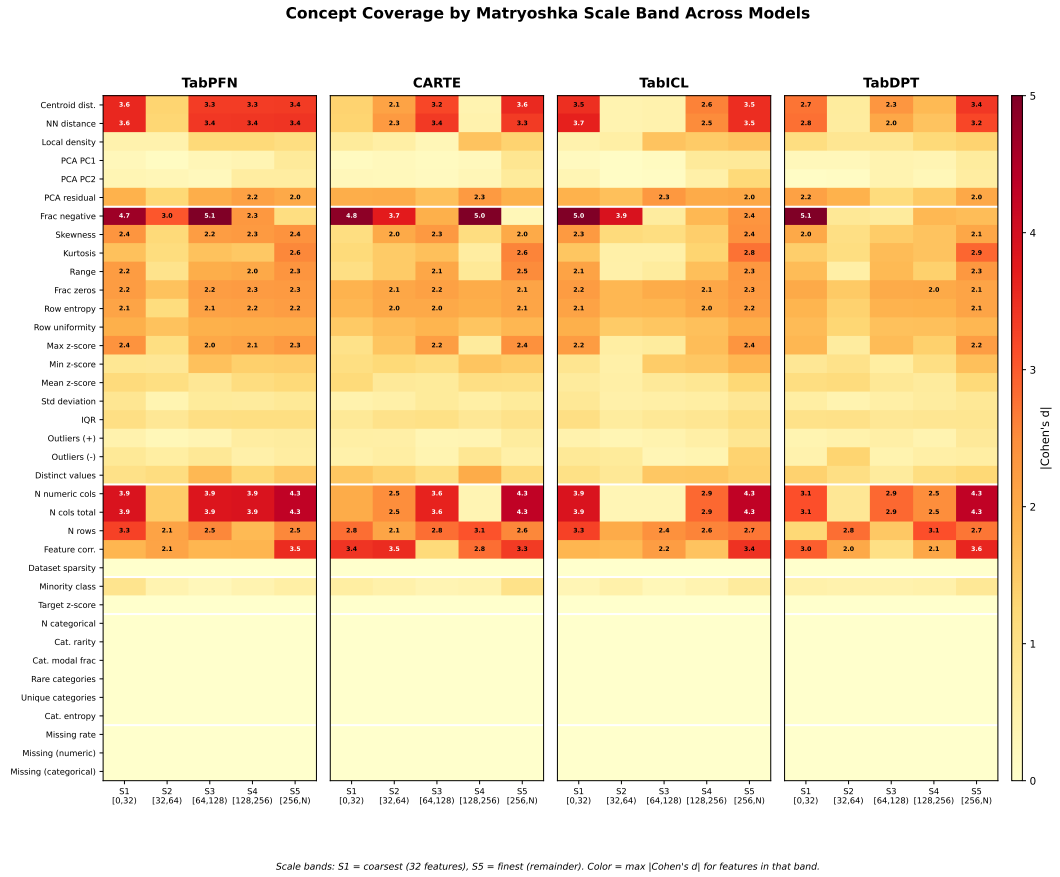


Figure 16: Concept coverage by Matryoshka scale band across four models. Cell color indicates the maximum |Cohen's d| achieved by any alive concept in that band for the given meta-concept. All models share the same coverage gaps (categorical, missing, target) and the same well-covered concepts (geometric, distributional, dimensional), differing primarily in the scale bands at which concepts are most strongly represented.

Feature	Patch column	Original value	Patched value	Original act.	Pa
f_{11}	checking_status	$0 \leq \cdots < 200$ DM	< 0 DM	1.437	
f_6	duration_months	15	18	3.369	
f_{92}	credit_history	critical account / other credits existing	existing credits paid duly till now	5.465	
f_{36}	duration_months	15	6	4.493	
f_{86}	checking_status	no checking account	< 0 DM	11.217	

Table 15: Single-column credit-g patches that suppress Mitra SAE concept activation for the focal credit-g row shown in Figure 1. The same concept index can fire on different input columns across rows; cohort-level patches for credit-g are reported in Table 17.

TabPFN and Mitra; despite flat observational firing frequencies (Appendix H), presence and importance are dissociated. Intervention curves (panels (c) and (d)) decay steeply within the top ten concepts, confirming that a small subset of unmatched concepts drives the per-row gap.

Unmatched concepts are sufficient. Restricting ablation to unmatched concepts produces nearly identical results to ablating *all* concepts, with negligible difference. This confirms that the performance gap between models is primarily driven by concepts unique to the strong model, not by differential use of shared concepts. The matched (shared) concepts contribute minimally to closing the gap—they are genuinely shared in both representation and function.

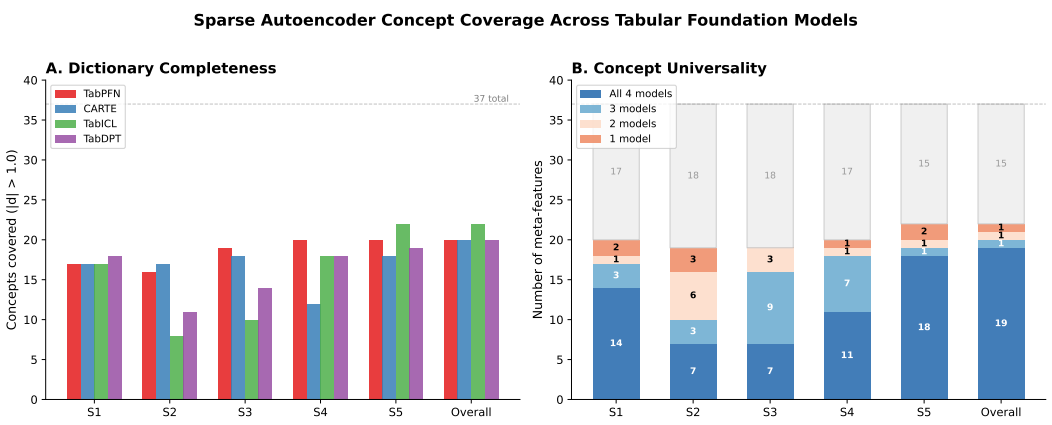


Figure 17: **(A)** Dictionary completeness: number of meta-concepts covered ($|d| > 1.0$) by each model's SAE at each Matryoshka scale band. **(B)** Concept universality: how many models share each covered meta-concept. 15 of 20 covered concepts are detected by all four models.

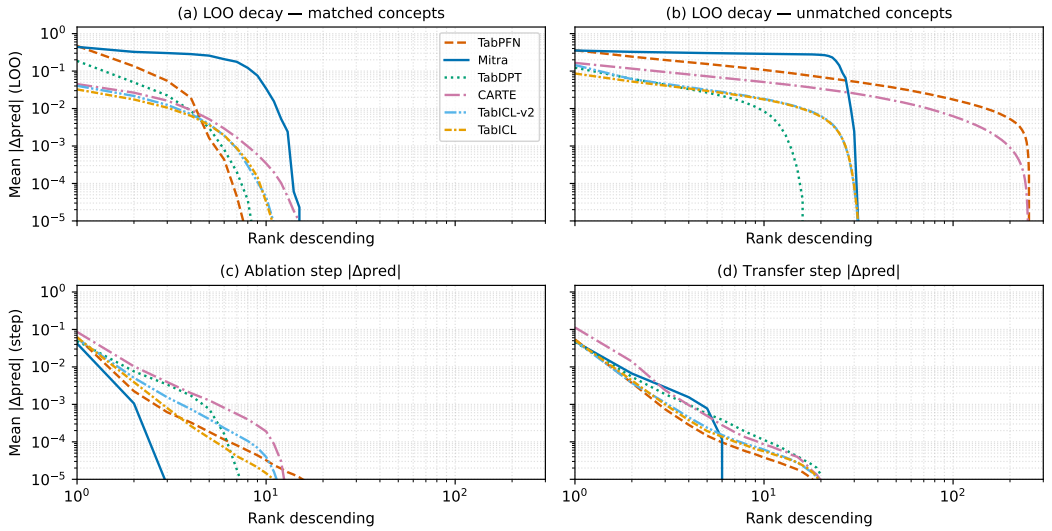


Figure 18: Row-averaged per-concept $|\Delta\text{pred}|$ decay for matched (a), unmatched (b), ablation (c), and transfer (d). Intervention curves (c, d) show steep decay within the top ten concepts, while leave-one-out curves have longer tails. A small number of unmatched concepts account for the majority of the performance gap, with diminishing returns beyond rank five.

Band distribution of selected concepts. The greedy ablation search disproportionately selects concepts from the lowest Matryoshka bands. Band 0 concepts are selected 6–10× more often than their share of the dictionary, while Bands 3–4 are selected 0.1–0.5× their share. This indicates that the concepts differentiating models are not fine-grained specializations in the higher bands, but core representational concepts.

Which features does ablation select? Table 16 shows the top 5 unmatched features most often selected when each weak model is ablated against every stronger model, for classification tasks; Table 20 in Appendix E gives the regression counterpart. The ablation-selected features split cleanly by task type: for every weak model, its classification and regression top features are disjoint. TabICL-v2 `f111` appears as a top deficit against CARTE and TabDPT in classification, while TabPFN `f5` is the top regression deficit feature against CARTE and TabICL-v2 (and a top-5 deficit against Mitra and TabDPT).

Weak	#1	#2	#3	#4	#5
CARTE	TabICL-v2$^{15}_{111}$	TabICL-v2$^{12}_{35}$	TabICL$^{12}_{30}$	TabICL$^{11}_{36}$	TabICL-v2$^{10}_{95}$
Mitra	TabPFN$^{10}_{40}$	TabICL-v2$^{8}_{111}$	TabPFN$^{8}_{12}$	TabICL-v2$^{7}_{35}$	TabICL-v2$^{7}_{95}$
TabDPT	TabICL-v2$^{6}_{111}$	TabICL-v2$^{5}_{24}$	TabICL-v2$^{5}_{108}$	TabPFN$^{6}_{40}$	TabPFN$^{6}_{33}$
TabICL	TabICL-v2$^{3}_{0}$	TabICL-v2$^{3}_{108}$	TabPFN$^{5}_{33}$	TabDPT$^{6}_{17}$	TabICL-v2$^{3}_{87}$
TabICL-v2	TabICL$^{5}_{61}$	TabDPT$^{7}_{17}$	TabICL$^{4}_{30}$	TabDPT$^{6}_{12}$	TabICL$^{4}_{44}$
TabPFN	TabICL-v2$^{6}_{24}$	TabICL-v2$^{5}_{107}$	TabICL-v2$^{4}_{0}$	TabICL-v2$^{4}_{3}$	TabICL$^{5}_{84}$

Table 16: **Per-weak-model ablation deficit profile, classification.** Top 5 unmatched features most often selected in ablation when each weak model is compared against every stronger model, ranked by selection frequency within each row. Cells follow $\mathrm{Strong}^{\%}_{\text{feat}}$: base text identifies the strong model that owns the feature, superscript is the selection frequency as $\%_{\text{S-losses}}$ (fraction of rows where the strong model beat the weak model and the feature appeared in the ablation set), subscript is the feature index.

Rank	Feature	Patch	Orig. act.	Patch act.	Mitra Δ	CARTE Δ
1	f_{92}	foreign_worker: no → yes	7.324	0.000	0.226	0.002
2	f_{63}	foreign_worker: no → yes	3.460	0.000	0.226	0.002
3	f_{54}	personal_status_sex: male: single → male: married/widowed	2.631	0.000	0.000	0.014
4	f_{6}	foreign_worker: no → yes	4.771	0.000	0.226	0.002
5	f_{98}	employment_since: ≥ 7 years → < 1 year	8.053	0.000	0.059	0.001
6	f_{34}	no representative row produced a qualifying patch	–	–	–	–
7	f_{11}	age_years: 36 → 30	1.805	0.000	0.031	0.007
8	f_{46}	age_years: 36 → 32	7.614	0.000	0.031	0.007
9	f_{95}	employment_since: 1 <= ... < 4 years → 4 <= ... < 7 years	7.924	0.000	0.000	0.000
10	f_{93}	credit_amount: 2301 → 448	2.184	0.000	0.121	0.012

Table 17: Weak-stable causal patch explanations for the credit-g panel. Patches are computed across the weak-stable cohort and may surface different input-column drivers than the focal-row patches in Table 15: a polysemantic concept can fire on different inputs across rows.

F.3 Input-level patch explanations

For the three datasets in Figure 3, we ground each top unmatched concept in a column-value perturbation: starting from a row where the concept fires, we search for the smallest edit that drops the concept's activation toward zero while having minimal impact on the weak model's prediction. Tables 17–19 list these patches alongside the resulting change in each model's prediction, identifying the input-level signal that distinguishes Mitra's representation from CARTE's.

F.4 Regression Gap Index

Table 20 is the regression counterpart to Table 16. TabICL is omitted because its public release has no regression variant.

F.5 Ablation selection bias

Greedy ablation selects features in order of $|\Delta\text{pred}|$ when the feature is zeroed, and this delta scales with the product of activation magnitude and decoder norm. The top unmatched features for TabPFN, CARTE, TabICL-v2, and TabDPT all have firing rates above 80%; only Mitra's top unmatched feature (`f8`, 16% firing) is genuinely selective. Ablation's top rankings thus surface the always-active, high-norm directions each model relies on by construction—core representational components rather than sparse concept detectors. This is consistent with the §4.2 finding that gap-closing concepts are core directions: the magnitude-dominated ranking is precisely why removing a small number of these concepts collapses the prediction gap. Despite observational firing rates of 22–43% (Table 29), no model's top-1 ablation-selected feature exceeds 7.5% of its winning rows,

Rank	Feature	Patch	Orig. act.	Patch act.	Mitra Δ	CARTE Δ
1	f_{22}	Curricular_units_2nd_sem_approved: 0 → 5; Age_at_enrollment: 27 → 21	14.128	0.000	0.926	0.123
2	f_{81}	Curricular_units_2nd_sem_enrolled: 4 → 6; Curricular_units_1st_sem_enrolled: 9 → 6; Curricular_units_2nd_sem_credited: 4 → 0; Curricular_units_1st_sem_credited: 5 → 0	8.027	2.631	0.115	0.030
3	f_{7}	Curricular_units_1st_sem_enrolled: 6 → 5	2.973	0.000	0.027	0.005
4	f_{86}	Curricular_units_2nd_sem_approved: 0 → 3	2.577	0.000	0.211	0.005
5	f_{47}	Curricular_units_1st_sem_approved: 6 → 12	2.194	0.000	0.627	0.013
6	f_{52}	Curricular_units_1st_sem_grade: 13 → 0	9.861	4.498	0.194	0.016
7	f_{19}	Curricular_units_2nd_sem_approved: 0 → 6	11.229	0.000	0.392	0.015
8	f_{29}	Mothers_qualification: Basic Education 2nd Cycle → Basic Education 3rd Cycle	2.349	0.000	0.000	0.000
9	f_{314}	Application_mode: 2nd phase - general contingent → 1st phase - general contingent	4.425	0.000	0.002	0.000
10	f_{88}	Curricular_units_2nd_sem_approved: 0 → 4	15.440	0.000	0.305	0.009

Table 18: Weak-stable causal patch explanations for the students dropout panel.

Rank	Feature	Patch	Orig. act.	Patch act.	Mitra Δ	CARTE Δ
1	f_{43}	Population: 2056 → 1659	7.949	0.000	0.109	0.129
2	f_{40}	Latitude: 34.23 → 32.8	6.618	2.394	0.166	0.033
3	f_{28}	Longitude: -119.78 → -119.02	2.878	0.000	0.076	0.000
4	f_{108}	Population: 620 → 2688	7.127	0.000	0.114	0.066
5	f_{2}	Longitude: -117.84 → -116.86	8.477	0.000	0.459	0.000
6	f_{12}	Longitude: -117.84 → -118.42	2.761	0.000	0.194	0.000
7	f_{14}	Longitude: -121.89 → -122.31	4.683	0.000	0.402	0.000
8	f_{105}	Latitude: 37.6 → 37.82	2.584	0.000	0.189	0.001
9	f_{17}	no representative row produced a qualifying patch	–	–	–	–
10	f_{64}	Latitude: 37.6 → 37.97	2.671	0.000	0.208	0.001

Table 19: Weak-stable causal patch explanations for the houses panel.

and no top-5 exceeds 32.9% (Figure 19A) — confirming that high firing frequency and causal importance are dissociated.

Although these features have high firing rates, they are not selected for every row: the top-1 ablation pick appears in only 3–7.5% of a model's winning rows, and the top-5 in 15–33% (Figure 19A). High firing rate measures how often a concept *can* fire, not how often the greedy selector picks it. The per-row selection pattern is structured: the same concept index repeatedly appears as the top deficit across different weak/strong model pairs (e.g., TabPFN f_5 in regression, TabICL-v2 f_{111} in classification; Tables 16–20), input-level perturbations that null a selected concept correspond to interpretable column edits (§F.3), and random-SAE controls fall well short of trained performance (§G). The picture is therefore high-norm core directions whose row-level selection carries identifiable semantic content, rather than a fixed set of always-active features.

F.6 CKA PREDICTS ABLATION EFFICIENCY

CKA also predicts ablation *efficiency*—the number of concepts K needed to close the gap ($r = -0.74$, $p = 0.002$): high-CKA pairs share most of their concept vocabulary, so the few unmatched concepts each carry high causal weight. However, CKA does *not* predict how much of the gap is closed ($r = -0.19$, $p = 0.51$). Gap closed varies independently of CKA — whether a performance difference is fully explained by discrete SAE concepts depends on the pair, not their geometric similarity. This produces a causal chain: geometric similarity → concept overlap → how many concepts are needed, decoupled from how much of the gap is closed.

Weak	#1	#2	#3	#4	#5
CARTE	TabPFN^{25}_{5}	$\text{TabICL-v2}^{18}_{124}$	$\text{TabICL-v2}^{16}_{17}$	TabPFN^{16}_{27}	$\text{TabICL-v2}^{15}_{63}$
Mitra	$\text{TabICL-v2}^{12}_{46}$	TabPFN^{11}_{5}	TabPFN^{10}_{12}	TabPFN^{9}_{40}	TabPFN^{8}_{41}
TabDPT	$\text{TabICL-v2}^{13}_{46}$	$\text{TabICL-v2}^{9}_{95}$	$\text{TabICL-v2}^{9}_{63}$	$\text{TabICL-v2}^{8}_{124}$	TabPFN^{10}_{5}
TabICL-v2	TabPFN^{11}_{5}	TabDPT^{4}_{108}	TabPFN^{7}_{19}	TabPFN^{7}_{12}	TabPFN^{5}_{16}
TabPFN	$\text{TabICL-v2}^{8}_{46}$	$\text{TabICL-v2}^{8}_{95}$	$\text{TabICL-v2}^{5}_{59}$	$\text{TabICL-v2}^{5}_{17}$	$\text{TabICL-v2}^{5}_{108}$

Table 20: **Per-weak-model ablation deficit profile, regression.** Regression counterpart of Table 16. TabICL is omitted because its public release has no regression variant. Cell format and ranking follow the classification table.

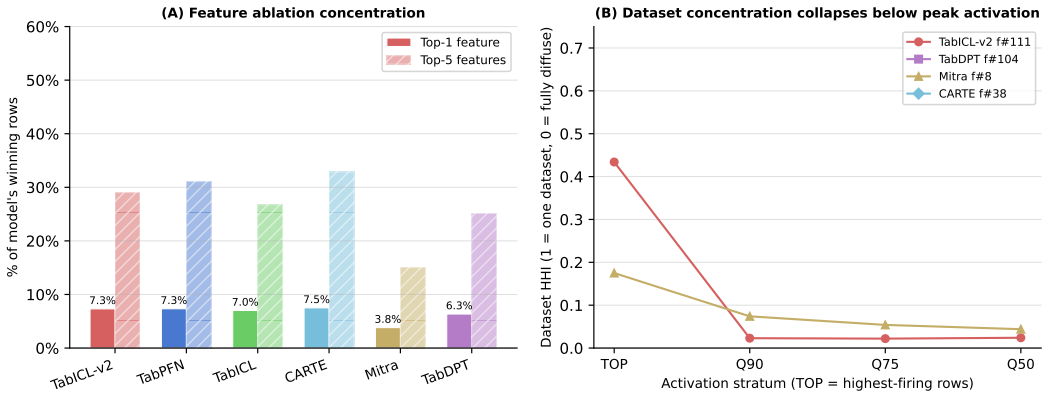


Figure 19: **(A)** Per-model cumulative ablation concentration: fraction of winning rows in which the top-1 and top-5 unmatched features appear in the ablation set. No model's top feature exceeds 7.5% of its winning rows; no top-5 exceeds 32.9%. Mitra (top-1 3.8%, top-5 15.0%) is the most diffuse. **(B)** Dataset Herfindahl–Hirschman Index (HHI) of activating rows across activation strata for four representative unmatched features. At the TOP stratum three of four features concentrate on a single dataset; at Q90 and below, all features are diffuse across 45–51 datasets.

F.7 Concept Explanatory Power Ranking

In head-to-head comparisons, TabICL-v2's unmatched concepts close more of the gap against all five other TFMs, followed by TabPFN, TabICL, and TabDPT (each winning against three of five). Mitra's unmatched concepts win only against CARTE. This ranking reflects concept explanatory power on the SAE test split — our strong/weak designations are local to each (pair, dataset) comparison and do not follow full benchmark rankings or SAE reconstruction quality (TabICL-v2 has among the lowest R^2).

F.8 Cross-Model Transfer

The transfer experiment requires a map between two models' embedding spaces, which we estimate from the concepts that *did* match. Each SAE decoder column is a unit direction in its model's embedding space, so an MNN-matched concept pair (i, j) supplies one paired direction: the decoder column of concept i in the strong model and the decoder column of concept j in the weak model. These matched decoder concepts act as landmarks between the two spaces. Before fitting, we run a leave-one-out sanity pass: for each landmark we refit ridge on the other $N - 1$ pairs and check the predicted target direction against the held-out target. Landmarks with negative LOO cosine are sign-flipped — a legitimate invariance, since the activation correlation matrix in Eq. 2 uses |act-sim| and discards sign — and anything still below threshold is dropped. We then fit a global linear map $M \in \mathbb{R}^{d_{\text{weak}} \times d_{\text{strong}}}$ by ridge regression ($\alpha = 1.0$, no intercept, since we are mapping directions rather than points) on the surviving landmark pairs. An unmatched concept's decoder direction in the strong model is passed through M to obtain a corresponding direction in the weak model's space; this mapped delta is injected into the weak model's cached hidden state at the extraction layer

across all sequence positions (context and query), and the weak model's remaining layers produce a modified prediction.

Landmark map quality. The cross-validated R^2 of the landmark map is consistently low. Across all 15 pairs and 687 (pair, dataset) runs, the mean $R^2 = -0.055$ and median $R^2 = -0.048$, with 12 of 15 pairs exhibiting a negative mean R^2 averaged across both directions; only the Mitra↔TabICL-v2 pair achieves $R^2 > 0.1$. Replacing the global ridge with a 2-layer MLP produces fits of comparable quality. Tightening the landmark filter from LOO cosine ≥ 0 to LOO cosine ≥ 0.20 (the mean random-SAE cosine) raises mean R^2 on well-determined pairs to $\approx +0.13$, but this is a metric effect rather than a functional one: the resulting map agrees with the deployed map on the highest-lift virtual atoms (median cosine 0.77–0.98 between the rank-0 greedy picks across six of seven directions tested). In other words, the multi-output R^2 absorbs reconstruction error across all 192–768 target dimensions, while transfer only requires the small subset of unmatched directions selected by greedy search to be stable. A low aggregate R^2 should therefore not be read as evidence that the map is unusable for transfer; it bounds the map's quality as a general-purpose translator, not its quality on the directions transfer actually exercises.

Semantic content vs. useful perturbations. A low aggregate R^2 leaves open whether transferred directions carry semantic content from the strong model or merely act as beneficial perturbations on the weak model's prediction. Two lines of evidence favor the former. First, random-SAE directions injected at matched magnitude close only 52% of the gap on average versus 90% for trained directions (Table 2; Appendix G); equivalently, trained directions deliver $\approx 3.4\times$ more gap closed per concept (0.10 vs. 0.030 gc/concept). Random directions of the same norm therefore account for roughly half of the closure; the remaining 38 points and the per-concept efficiency gap are the part attributable to concept content. Second, input-level patches (§F.3) ground individual transferred concepts in semantically coherent column-value perturbations (credit risk indicators in credit-g, academic trajectory in students_dropout, geographic pricing in houses), consistent with the directions encoding interpretable structure rather than arbitrary useful noise. Neither argument rules out the perturbation hypothesis decisively, but each is more consistent with semantic transfer.

Per-pair variance. Figure 20 breaks gap closed out by ordered (strong, weak) pair for both interventions. The per-strong-model summaries (Tables 1 and 2) flatten variance the heatmap surfaces: ablation underperforms uniformly when Mitra is strong (≈ 0.75 across all five partners), and transfer underperforms when CARTE is the weak target (gc 0.67–0.79) regardless of source.

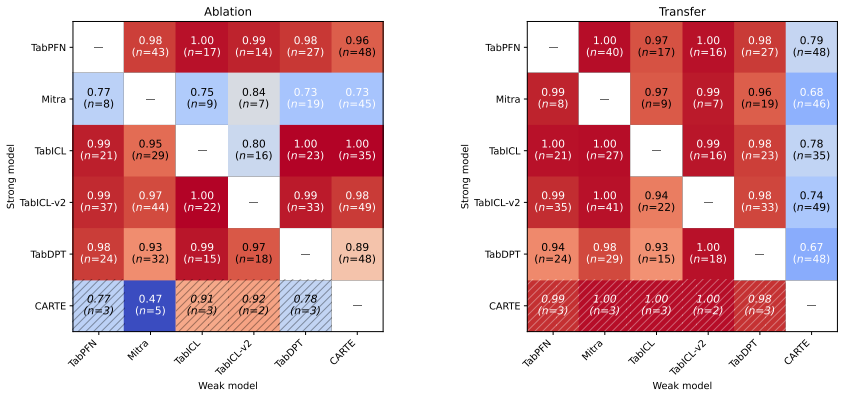


Figure 20: Mean gap closed for ablation (left) and transfer (right); rows = strong model, columns = weak model. Cells with $n < 5$ are italicized and hatched.

F.9 Off-manifold patch explanations

To test that the off-manifold transferred concepts correspond to interpretable input structure, we causally patch each one: for a concept the recipient lacks, we find the smallest edit to a donor row that suppresses that concept's activation, and measure the resulting change in the recipient's true-

Table 21: Attrition for causal patching of off-manifold transferred concepts.

Stage	N
Starting concept universe	335
never searchable: conflicting TabICL versions required	1
no qualifying patch found	3
CARTE as recipient, refit induces noise larger than effect	11
Patch with measured recipient prediction	320

Table 22: Patch character over the 320 concepts with a measured recipient prediction, by quartile of the concept's transfer rank. 'ext.' is the fraction whose activation the patch drove to exactly zero; histograms are the per-concept suppression over $[0, 1]$ in ten bins, common vertical scale. 'select.' is how many times more the target concept moved than any other concept at that row; 'centr.' is the patched row's centrality over its original. Medians throughout.

Quartile	transfer rank	N	suppression	ext.	cols	select.	centr.
Q1	rank ≤ 3	86		62%	4.0	27.1	1.51
Q2	$3 <$ rank ≤ 9	76		26%	4.0	3.4	1.60
Q3	$9 <$ rank ≤ 34	81		46%	4.0	1.8	1.52
Q4	rank > 34	77		55%	3.0	1.9	1.63
All		320		48%	4.0	2.6	1.57

class probability. We take the 335 concepts that are both off-manifold (the share of their mapped direction outside the recipient's principal subspace, at 90% of variance, in $[0.6, 0.8]$) and robustly transferred (accepted on 200–499 rows); 320 yield a measured recipient prediction (Table 21). The 15 losses are measurement gaps rather than nulls: one concept needs mutually incompatible TabICL versions, three admit no qualifying patch, and eleven are CARTE-recipient, whose per-dataset refit moves predictions by more than the effect being read.

The patches genuinely and selectively suppress their target (Table 22). Suppression is bimodal, driving the concept's activation to exactly zero on 48% of concepts; for the highest-transfer-rank concepts (rank ≤ 3) the patch moves the target concept $27\times$ more than any other, over a median of four edited columns.

Table 23 summarizes what each recipient is missing and how much a single concept is worth. The per-concept effect is small for TabDPT: a median 2.6×10^{-3} in true-class probability, against 10.8–23.2×10^{-3} for the other recipients. TabDPT absorbs transferred structure successfully (§4.2) but spreads it across many co-active concepts, so no single concept is necessary and single-concept patching has little to grasp there: a signature of distributed encoding, not an exclusion. Tables 24–27 give example patches across the range of ablation effect, pairing each concept's ablation effect with what the input edit achieves.

F.10 Limitations of the Causal Methodology

Four caveats temper the ablation and transfer results in §4.2.

Greedy search. Both interventions select concepts greedily by per-row $|\Delta\text{pred}|$. This finds a sufficient subset that closes the gap but does not search the full combinatorial space, so the reported K (median 2 concepts ablated) is an upper bound on the minimal sufficient set.

Selection bias toward high-magnitude features. As detailed in §F.5, greedy ablation rankings surface always-active, high-norm directions rather than selective concept detectors. Top ablation features are therefore not interchangeable with "most interpretable concepts" — they are the directions whose removal produces the largest immediate prediction shift, which is exactly what closes the per-row gap.

Table 23: What each recipient is missing. Each cell counts the distinct off-manifold concepts a donor carries that the recipient lacks, and the median effect of ablating one, in true-class probability ($\times 10^{-3}$). Row and column totals count distinct concepts, so they are smaller than the sums: a concept missing from two recipients is counted once in each column but once overall. TabICL and TabICL-v2 cannot share an environment, so that pair is unmeasured rather than empty.

Donor	mitra	tabdpt	tabpfn	tabicl	Distinct
mitra	–	8 / 9.0	3 / 9.6	5 / 13.6	8
tabdpt	51 / 6.8	–	16 / 5.4	18 / 17.4	58
tabicl	59 / 34.6	73 / 2.9	60 / 35.7	–	105
tabicl_v2	45 / 7.2	53 / 2.4	44 / 3.1	–	71
tabpfn	65 / 53.4	45 / 2.4	–	18 / 21.5	78
Distinct	220 / 23.2	179 / 2.6	123 / 10.8	41 / 18.8	320

Linear map quality bounds the transfer claim. The cross-space ridge map (§F.8) has cross-validated $R^2 \approx -0.05$ across 687 (pair, dataset) runs, and a 2-layer MLP fits no better. As argued in §F.8, the high-lift directions selected by greedy search are stable across map variants, and the per-pair evidence (random-SAE controls, input-level patch coherence) is more consistent with semantic transfer than with beneficial perturbation. We cannot rule out that some fraction of gap closed reflects perturbation rather than concept content, particularly for low-acceptance directions in the long tail.

Benchmark scope. All causal results are computed on TabArena's 51 datasets and inherit its sampling characteristics (domain mix, task-type distribution, dataset-size cap). Generalization to other benchmarks or to in-the-wild tabular data is untested.

We analyze six models spanning three architectural families; HyperFast and Tabula-8B are included in geometric comparisons but excluded from causal analysis due to architectural incompatibilities detailed in §3. The tabular foundation model landscape is evolving quickly, and models released since this study are not included. They do not change the diagnostic: the method needs only two models with a per-row performance gap and an extractable embedding, so each new model is additional pairs for the same pipeline. What may not carry over is the finding itself, in particular for architectures with fundamentally different representation formation (e.g., state-space models). A complete picture of convergence would require methods adapted to hypernetwork and LLM-based architectures.

G Random Baseline Comparisons

To verify that results throughout the pipeline reflect genuine structure learned by the SAEs rather than arbitrary projections in high-dimensional space, we repeat the full pipeline—matching, importance, and ablation—using randomly initialized SAEs with identical architectures but no training.

G.1 Matching

Random SAE concepts produce pairwise activation correlations of 0.12–0.28 (vs. 0.50–0.80 for trained), confirming that cross-model concept correspondence is a property of training, not architecture.

G.2 Matryoshka Bands

Randomly initialized SAEs show no band structure: variance distributes roughly proportionally to band size (5–7% in band 0 vs. 46–50% in band 4), in contrast to the sharp 66–84% concentration in band 0 for trained SAEs (Figure 21).

Table 24: Patch explanations, quartile Q1 of the ablation effect (largest; $|\Delta_{\text{ablate}}|$ from 28.5 to 578.7 $\times 10^{-3}$). Up to 3 concepts per dataset, drawn at random within the quartile and not selected for quality; no concept repeats across quartiles. A dataset contributes fewer rows, or none, where the quartile holds fewer of its concepts. Repair edits, which walk back collateral rather than suppress, are *greyed*. Prediction change is in true-class probability ($\times 10^{-3}$): what ablating the concept achieves, and what the input patch achieved.

Rows	Donor	Recip.	Patch	Act.	Ablate	Patch
splice						
1	tabicl f_{458}	mitra	position_-29: T → A position_-11: T → G position_-24: T → A *+10 more*	2.84 → 1.03	-200.0	-174.2
1	tabpfn f_{116}	mitra	position_-19: C → A position_-16: C → A position_-8: C → A *+4 more*	6.44 → 5.92	-111.7	-11.2
1	tabicl f_{26}	tabdpt	position_-21: A → G position_18: A → C position_-11: C → T *+10 more*	5.46 → 0.00	-80.8	-28.6
Amazon_employee_access						
2	tabicl f_{243}	tabpfn	ROLE_ROLLUP_1: 117902 → 117882 ROLE_FAMILY: 117887 → 118131 ROLE_ROLLUP_2: 117903 → 117883 *+1 more*	1.66 → 0.81	-281.2	-219.7
1	tabicl f_{130}	tabpfn	ROLE_DEPTNAME: 118783 → 118202	3.07 → 0.00	-43.0	-38.2
2	tabicl f_{228}	tabpfn	ROLE_ROLLUP_1: 117929 → 118006 ROLE_FAMILY: 117887 → 118762	1.85 → 0.00	-41.9	-73.1
E-CommereShippingData						
1	tabpfn f_{262}	mitra	Mode_of_Shipment: Ship → Flight Cost_of_the_Product: 281 → 280	2.11 → 1.53	-209.4	-198.0
4	tabpfn f_{337}	mitra	Weight_in_gms: 5209 → 5121 Cost_of_the_Product: 145 → 152 Prior_purchases: 2 → 6	6.11 → 5.18	+71.2	+9.5
5	tabdpt f_{152}	mitra	Customer_rating: 3 → 4 Gender: M → F	2.73 → 0.00	+28.7	+25.9
anneal						
1	tabpfn f_{140}	tabdpt	oil: not_applicable → Y temper_rolling: not_applicable → T blue_bright_varn_clean: B → not_applicable *+1 more*	2.54 → 1.05	-578.7	-409.7

G.3 Ablation

Repeating the full pipeline with randomly initialized SAEs isolates the learned signal (Figure 22). Trained SAEs close 93% of the gap vs. 58% for random, winning 96% of the 639 comparisons where the random arm completed (615 of 639). Each trained concept carries approximately 6× more information than a random direction (0.36 vs. 0.06 gc per concept). The elevated random baseline (0.58 mean gc) is driven by architectural sensitivity in certain models (Mitra, TabICL-v2), where even random embedding perturbations can shift predictions; this sensitivity equally inflates both trained and random results, preserving the relative advantage.

G.4 Transfer

We repeat the cross-model transfer experiment with randomly initialized SAEs in place of trained ones, holding the ridge-mapping pipeline, landmark filter, and greedy concept selection fixed. Across the same 687 (pair, dataset) runs reported in Table 2, trained-SAE transfer closes a mean of 90% of the gap while random-SAE transfer closes 52%. Random transfer also requires roughly

Table 25: Patch explanations, quartile Q2 of the ablation effect (upper-middle; $|\Delta_{\text{ablate}}|$ from 3.5 to 28.4 $\times 10^{-3}$). Up to 3 concepts per dataset, drawn at random within the quartile and not selected for quality; no concept repeats across quartiles. A dataset contributes fewer rows, or none, where the quartile holds fewer of its concepts. Repair edits, which walk back collateral rather than suppress, are *greyed*. Prediction change is in true-class probability ($\times 10^{-3}$): what ablating the concept achieves, and what the input patch achieved.

Rows	Donor	Recip.	Patch	Act.	Ablate	Patch
splice						
4	mitra f_{107}	tabdpt	position_21: T → A position_-5: G → A position_-6: A → C *+12 more*	3.51 → 2.03	-10.2	-2.9
15	tabicl f_{213}	tabdpt	position_-25: A → C position_-3: A → T position_-1: G → C *+1 more*	5.25 → 4.09	+9.4	+3.3
2	tabicl f_{147}	tabdpt	position_25: T → G position_-3: T → G position_4: A → C *+1 more*	2.80 → 0.00	+4.1	-4.0
Amazon_employee_access						
1	tabicl f_{40}	tabpfn	ROLE_DEPTNAME: 118783 → 120016 RESOURCE: 34716 → 77300	5.52 → 2.22	-28.3	-18.3
4	tabdpt f_{78}	mitra	ROLE_FAMILY_DESC: 239652 → 240983 MGR_ID: 7389 → 2710 ROLE_TITLE: 118747 → 118702 *+1 more*	7.94 → 0.00	-19.5	-19.4
16	tabdpt f_{165}	mitra	ROLE_FAMILY: 249618 → 118453 *ROLE_TITLE: 273308 → 259173* *RESOURCE: 40237 → 39518*	3.71 → 0.00	-7.7	-7.7
E-CommereShippingData						
2	tabpfn f_{179}	mitra	Mode_of_Shipment: Ship → Road Customer_rating: 5 → 3 Product_importance: medium → low *+1 more*	2.31 → 1.67	+25.2	+17.9
14	tabdpt f_{33}	mitra	Weight_in_gms: 1384 → 1336 Customer_care_calls: 5 → 6	2.20 → 1.56	-23.1	-6.7
1	tabpfn f_{170}	tabicl	Cost_of_the_Product: 219 → 278 Customer_care_calls: 5 → 7 Discount_offered: 3 → 6	4.27 → 3.42	-9.7	-2.7
anneal						
4	tabicl f_{808}	tabdpt	hardness: 80 → 76 thick: 1.6 → 0.7604 width: 1320 → 1293 *+2 more*	4.00 → 1.21	-3.8	-10.8
1	tabpfn f_{161}	tabdpt	lustre: not_applicable → Y surface-quality: G → F hardness: 80 → 81	3.01 → 2.47	-3.6	-3.2
18	tabicl f_{448}	tabdpt	thick: 0.6 → 1.078 surface-quality: F → E	2.39 → 0.00	-3.5	-3.5

twice as many concepts to reach this lower ceiling (mean $K_R = 17.3$ vs. $K = 8.9$), so per-concept yield drops from ≈ 0.10 to ≈ 0.030 gap closed — a $\approx 3.4\times$ advantage for trained directions. As with ablation, the non-trivial random baseline reflects architectural sensitivity to embedding-space perturbations rather than evidence that the trained map is no better than noise; the relative gap between trained and random is the signal that semantically aligned concepts, not arbitrary directions, drive transfer success. This analysis is the empirical basis for the "semantic content vs. useful perturbations" argument in §F.8.

Table 26: Patch explanations, quartile Q3 of the ablation effect (lower-middle; $|\Delta_{\text{ablate}}|$ from 1.2 to 3.4 $\times 10^{-3}$). Up to 3 concepts per dataset, drawn at random within the quartile and not selected for quality; no concept repeats across quartiles. A dataset contributes fewer rows, or none, where the quartile holds fewer of its concepts. Prediction change is in true-class probability ($\times 10^{-3}$): what ablating the concept achieves, and what the input patch achieved.

Rows	Donor	Recip.	Patch	Act.	Ablate	Patch
splice						
4	tabicl_v2 f_{381}	tabdpt	position_-20: G → T position_3: G → A position_-12: A → C	1.20 → 0.00	-2.6	-2.4
10	tabicl f_{51}	tabdpt	position_-30: G → T position_21: G → T position_-21: T → C *+7 more*	4.03 → 0.00	-2.0	-2.2
1	tabicl f_{360}	tabdpt	position_-30: C → G position_-9: A → T position_29: G → T *+7 more*	5.10 → 0.00	-1.2	-1.4
Amazon_employee_access						
14	tabdpt f_{177}	tabpfn	RESOURCE: 28510 → 42085	1.81 → 0.00	-2.4	-2.0
6	tabicl f_{105}	mitra	ROLE_TITLE: 118826 → 118054 RESOURCE: 45325 → 42085 ROLE_FAMILY_DESC: 257115 → 260942 *+2 more*	8.16 → 0.00	-2.4	-2.9
15	tabdpt f_{67}	mitra	ROLE_FAMILY_DESC: 240983 → 124151 ROLE_ROLLUP_2: 118300 → 118052	3.44 → 0.00	-2.0	-2.9
anneal						
1	tabicl f_{357}	tabdpt	len: 4880 → 635 bw_me: not_applicable → B	1.90 → 0.00	-3.4	-3.5
5	tabicl f_{198}	tabdpt	width: 609.9 → 50.72 len: 301 → 468 steel: not_applicable → V	4.05 → 1.47	-2.5	-2.5
6	tabicl f_{177}	tabdpt	bt: not_applicable → Y width: 385.1 → 634.5 bw_me: M → not_applicable	4.29 → 0.00	-1.5	-2.1

H Unmatched Concept Structure

Cross-model concept correspondence. The MNN matching pipeline identifies 846 multi-model concept groups across 9,398 alive SAE concepts from six models, plus 2,073 unmatched concepts that remain as singletons unique to a single model. 45 concept groups are *universal*—present in all six models—and the remaining multi-model groups are distributed across 2–5 model participations.

Table 28 summarizes the pairwise matching statistics. The fraction of each model's alive concepts that match to at least one concept in another model ranges from 27.8% (TabICL) to 43.4% (Mitra). Trained SAE concepts produce mean match correlations of 0.32–0.62, compared to 0.12–0.29 for randomly initialized SAEs (Appendix G), confirming that cross-model concept correspondence is learned, not architectural. The count of unmatched concepts is not the whole story: the Matryoshka SAE architecture concentrates 47–77% of reconstruction variance in the lowest band (Appendix Figure 15), so *where* in the dictionary a concept is missing matters more than how many are missing.

Unmatched features fire broadly on winning rows. For each model pair and dataset, we identify the rows where the stronger model outperforms the weaker (using AUC for classification, RMSE for regression) and count how often each of the stronger model's *unmatched* features fires (activation > 0) on those rows. This purely observational analysis requires no causal intervention—it asks only which unmatched features are *present* when one model wins.

Table 27: Patch explanations, quartile Q4 of the ablation effect (smallest; $|\Delta_{\text{ablate}}|$ from 0.0 to 1.2 $\times 10^{-3}$). Up to 3 concepts per dataset, drawn at random within the quartile and not selected for quality; no concept repeats across quartiles. A dataset contributes fewer rows, or none, where the quartile holds fewer of its concepts. Prediction change is in true-class probability ($\times 10^{-3}$): what ablating the concept achieves, and what the input patch achieved.

Rows	Donor	Recip.	Patch	Act.	Ablate	Patch
splice						
4	tabicl f_{204}	tabdpt	position_-20: G → A position_-5: G → T position_25: T → C *+3 more*	3.12 → 0.00	-0.5	-0.5
1	tabpfn f_{350}	tabdpt	position_11: C → G position_2: T → G position_-10: G → T *+11 more*	1.38 → 0.00	-0.5	-0.5
5	tabicl f_{238}	tabdpt	position_-28: C → G position_-11: G → A position_-10: T → G *+6 more*	4.36 → 0.00	-0.0	-0.0
Amazon_employee_access						
6	tabdpt f_{192}	mitra	ROLE_TITLE: 117905 → 118523	2.64 → 0.00	+0.6	+1.8
1	tabicl_v2 f_{390}	mitra	RESOURCE: 81298 → 6977 ROLE_DEPTNAME: 118910 → 118896	2.74 → 0.72	+0.4	+1.5
5	tabicl_v2 f_{173}	tabpfn	RESOURCE: 25993 → 16018 ROLE_ROLLUP_2: 119070 → 118991	1.11 → 0.00	-0.3	-0.7
E-CommereShippingData						
2	tabpfn f_{102}	mitra	Prior_purchases: 4 → 2 Customer_care_calls: 6 → 4 Mode_of_Shipment: Ship → Road *+2 more*	2.53 → 1.02	-0.0	-160.0
anneal						
4	tabicl f_{395}	tabdpt	bf: not_applicable → Y condition: S → A bw_me: not_applicable → M	3.78 → 0.00	-0.8	-0.2
11	tabicl_v2 f_{415}	tabdpt	shape: COIL → SHEET formability: 3 → 5 width: 609.9 → 1234	1.53 → 0.00	+0.5	+0.1
16	tabicl_v2 f_{219}	tabdpt	bw_me: not_applicable → M packing: 2 → not_applicable width: 610 → 23.07 *+1 more*	2.32 → 0.00	+0.2	-0.3

Table 28: Per-model SAE summary. *Alive*: concepts that fire on at least one test row. *MNN %*: mutual nearest neighbor matches above the per-pair p90 noise floor. *Threshold %*: additional features with a best correlate above the noise floor (one-sided). *Unmatched %FVE*: activation variance in unmatched concepts—the ceiling for ablation.

	TabPFN	Mitra	TabICL	TabICL-v2	TabDPT	CARTE
Emb dim	192	512	512	512	768	300
Alive	634	1,686	1,553	2,004	2,436	1,085
MNN %	28.4	43.4	27.8	33.1	39.9	34.6
Threshold %	22.6	36.4	58.7	41.0	45.4	46.5
Unmatched %FVE	64.6	8.6	0.9	20.1	28.3	23.0

The result is strikingly flat (Table 29): thousands of unmatched features fire on winning rows, and the most-frequently-firing features are essentially tied. In classification, four of six weak models share the same top-3 unmatched features—TabICL-v2 `f45`, `f58`, and `f106`—each firing on 22–43% of

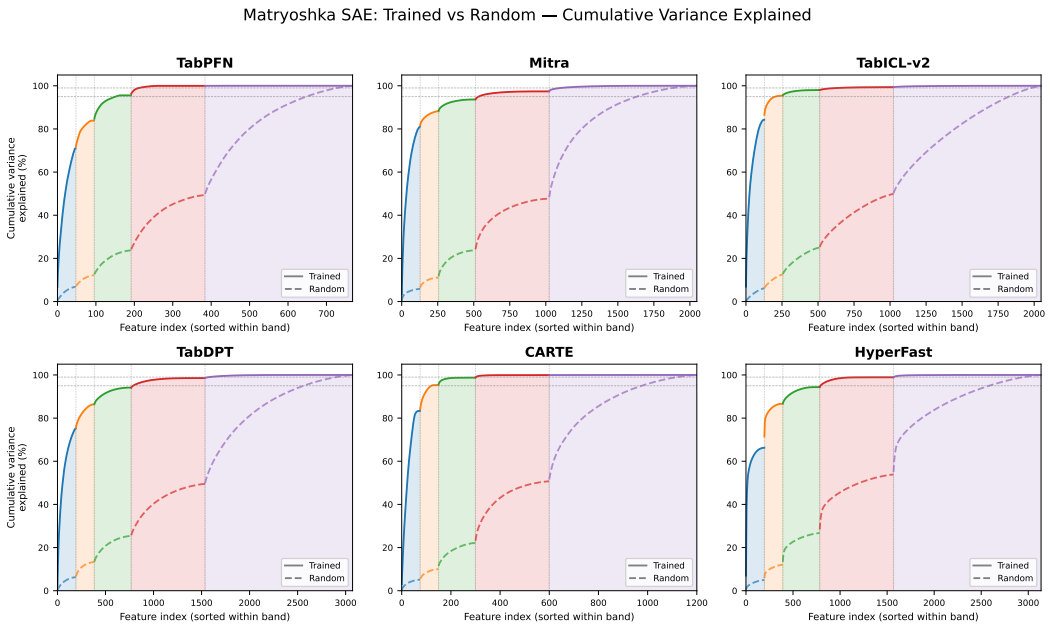


Figure 21: Trained (solid) vs. random (dashed) SAE cumulative variance explained by Matryoshka band. Trained SAEs concentrate variance in early bands; random SAEs distribute uniformly.

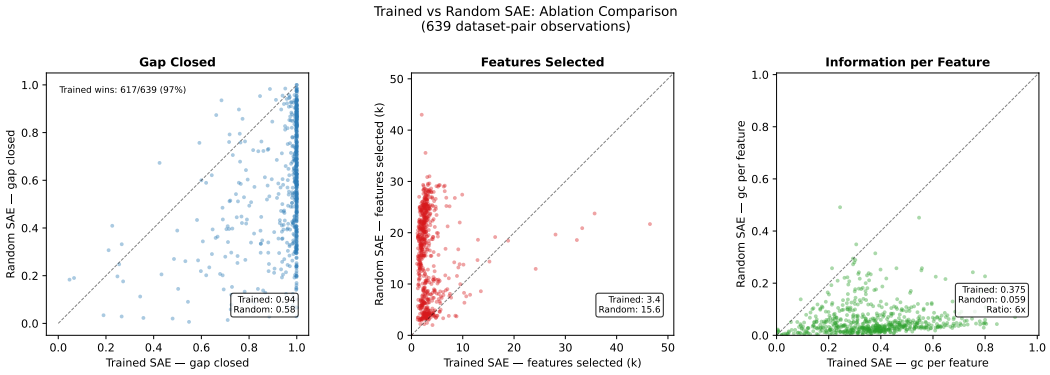


Figure 22: Trained vs. random SAE ablation comparison across 639 dataset-pair observations. **Left:** Gap closed—trained SAEs dominate (96% win rate). **Center:** concepts selected—random SAEs require nearly 4× more concepts. **Right:** Information per concept—trained concepts carry 6× more signal per direction.

lost rows depending on the weak model. In regression, the dominant strong model shifts (TabPFN features top the list against Mitra and CARTE, while TabICL-v2 features top the list against TabPFN and TabDPT), but within each cell the top features remain tied. No single unmatched feature stands out by firing frequency alone. Presence of unmatched features on winning rows is diffuse, not concentrated in a small set of critical concepts.

Co-firing reveals structure despite flat individual rates. Although individual firing frequencies are flat, *pairwise* co-firing is not. Figure 23 aggregates co-firing by Matryoshka band: for each band pair (b_i, b_j), it reports, for each band pair, the mean co-firing rate on winning rows and the combined reconstruction variance. The cell color encodes the mean pair FVE—the combined reconstruction variance of co-firing feature pairs from those bands, computed from per-feature activation variance.

		Classification			Regression		
Weak model	n_{lost}	Strong	Feature	Freq	Strong	Feature	Freq
CARTE	25,297	TabICL-v2	f45	21.9%	TabPFN	f0	26.2%
		TabICL-v2	f58	21.9%	TabPFN	f1	26.2%
		TabICL-v2	f106	21.9%	TabPFN	f5	26.2%
Mitra	17,528	TabICL-v2	f45	29.0%	TabPFN	f0	34.7%
		TabICL-v2	f58	29.0%	TabPFN	f1	34.7%
		TabICL-v2	f106	29.0%	TabPFN	f5	34.7%
TabDPT	10,173	TabICL-v2	f45	31.4%	TabICL-v2	f0	47.0%
		TabICL-v2	f58	31.4%	TabICL-v2	f3	47.0%
		TabICL-v2	f106	31.4%	TabICL-v2	f22	47.0%
TabPFN	8,490	TabICL-v2	f45	42.8%	TabICL-v2	f0	49.2%
		TabICL-v2	f58	42.8%	TabICL-v2	f23	49.2%
		TabICL-v2	f106	42.8%	TabICL-v2	f34	49.2%
TabICL	6,239	TabICL-v2	f39	35.9%		—	
		TabICL-v2	f45	35.9%		—	
		TabICL-v2	f58	35.9%		—	
TabICL-v2	4,439	TabICL	f16	39.1%	TabDPT	f4	60.4%
		TabICL	f22	39.1%	TabDPT	f121	59.3%
		TabICL	f29	39.1%	TabDPT	f52	58.8%

Table 29: Top-3 most frequently firing unmatched features on winning rows, per weak model and task type, sorted by n_{lost} descending. *Freq* = fraction of rows the weak model lost on which the feature fires (activation > 0). No ablation ranking or importance weighting — purely observational presence. TabICL is classification-only.

The pattern differs sharply across models. TabPFN's band-0 features co-fire on 95% of winning rows (mean pair FVE 2.9%), forming a tightly coupled block that activates jointly. TabICL-v2 and CARTE show similar saturation in band 0 (67–75% co-firing). By contrast, Mitra's band-0 features co-fire on only 2% of winning rows despite carrying 79% of reconstruction variance—they fire independently, not as a block. TabDPT is intermediate at 14%. The co-firing gradient from band 0 to band 4 is steep for all models, confirming that the Matryoshka band structure organizes not only reconstruction importance but also activation coupling.

This pairwise structure is the bridge between the flat individual firing rates and the causal analysis in Section 4.2: co-firing bands identify *coupled blocks* of unmatched features that a weak model is missing simultaneously, and ablation will test whether removing those blocks from the strong model degrades its predictions.

Unmatched features are model-private. The top unmatched features of different models fire on nearly disjoint rows. Across all 15 model pairs, the highest observed Spearman correlation between any top-5 unmatched feature of one model and any top-5 unmatched feature of another is $|\rho| = 0.12$ (TabPFN f12 vs. TabDPT f104, $n = 8{,}907$); the maximum top-1% Jaccard overlap for the same pair is 0.092. The surviving pair overlap traces to the wine_quality dataset; excluding that dataset reduces both metrics toward zero. Each model's unmatched directions are functionally dissociated at the row level—different models win on different rows using different features. We note that Spearman and Jaccard are rank-based summaries that cannot rule out shared geometry invisible to marginal correlation; a full representational similarity analysis would be needed to confirm that.

Peak activations are concentrated. The rows where a feature fires most strongly are not representative of that feature's activation distribution. Figure 19B reports dataset concentration (Herfindahl–Hirschman Index; HHI $= 1$ means all rows from a single dataset) across activation strata for four representative unmatched features. At the TOP stratum TabDPT f104 draws 82% of its rows from wine_quality (HHI $= 0.68$), and TabICL-v2 f111 draws 64% from the same dataset (HHI $=$ 0.43). At the Q90 stratum, both features collapse to benchmark-wide diffusion: TabDPT f104 top-dataset share falls to 5.4% (HHI ≈ 0.03) and TabICL-v2 f111 to 3.8% (HHI ≈ 0.02). Only Mitra f8 retains partial concentration beyond the peak (15% at Q90, decaying to 11.8% at Q75 before collapsing), consistent with its low unmatched variance (Table 28); even this feature covers 23 of

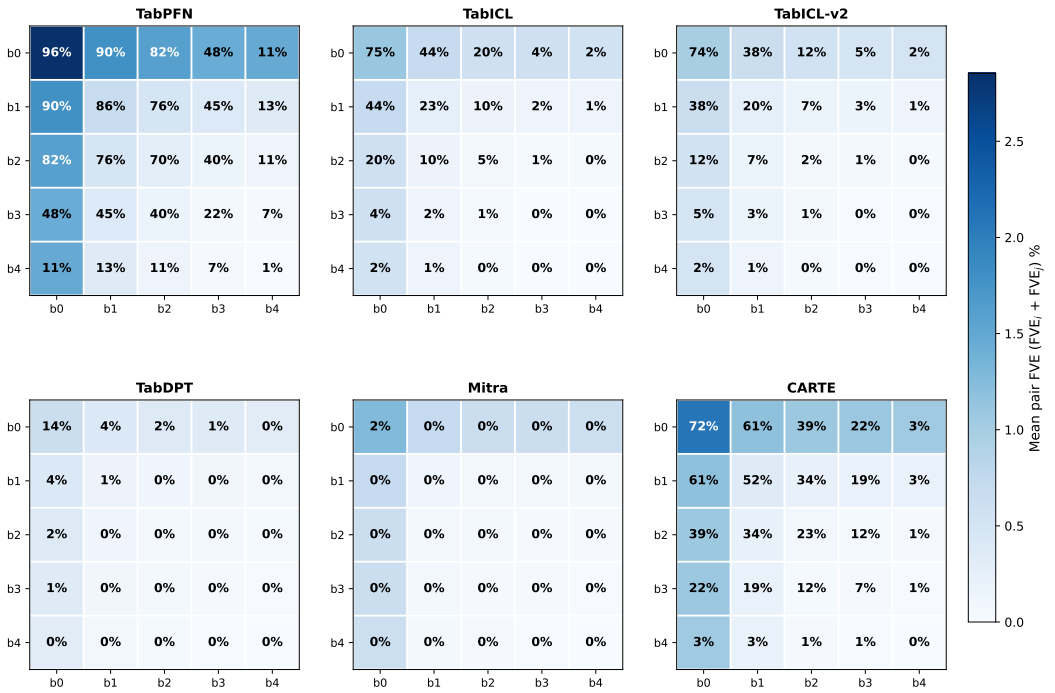


Figure 23: Pairwise co-firing of unmatched features by Matryoshka band, averaged across all weak opponents (classification). Cell color = mean pair FVE (combined per-feature reconstruction variance of co-firing pairs from those bands). Text = percentage of winning rows where both features fire simultaneously. TabPFN and CARTE show tightly coupled band-0 blocks (95% and 68% co-firing); Mitra's band-0 features fire independently (2%) despite carrying the highest reconstruction variance (79% FVE).

51 datasets at Q75. Developing a one-sentence label that summarizes the concept therefore should draw exemplar rows from across the datasets in order to generalize.

Classification and regression recruit different unmatched features. The task-type split in Table 29 is systematic (for models evaluated on both task types): top-firing unmatched features in classification and regression are disjoint. Classification is dominated by TabICL-v2 features (`f45`, `f58`, `f106`) across four of six weak models, while regression is split between TabPFN features (against Mitra and CARTE) and TabICL-v2 features (against TabPFN and TabDPT). No feature appears in both task-type columns for the same weak model. This disjointness holds at the observational level, it does not depend on ablation ranking or importance weighting.

CKA predicts concept overlap. Geometric alignment measured by CKA predicts concept sharing: the number of shared concept groups between a pair of models correlates with their CKA ($r = 0.76$, $p = 0.001$). Concept groups spanning all six models are present in every pair's unmatched pool (100% incidence), while pair-specific concepts appear in only 56.5% of pools, confirming that shared concepts are more structurally stable than model-private ones. 90% of the most frequently occurring unmatched concepts are model-driven rather than dataset-driven—the top two strong models account for >80% of each concept's appearances—confirming that concept structure reflects architectural properties, not benchmark artifacts.

I Complete Intervention Results

Each page shows the ablation grid for one dataset. Each panel corresponds to one (weak model, strong model) pair; the x-axis is gap closed (0 = no improvement, 1 = full gap eliminated) and the y-axis is the number of concepts ablated. Classification grids show all 15 pairs (3×5); regression

grids show 10 pairs (2×5), as TabICL is classification-only. Datasets are listed alphabetically within each task type.

I.1 CLASSIFICATION

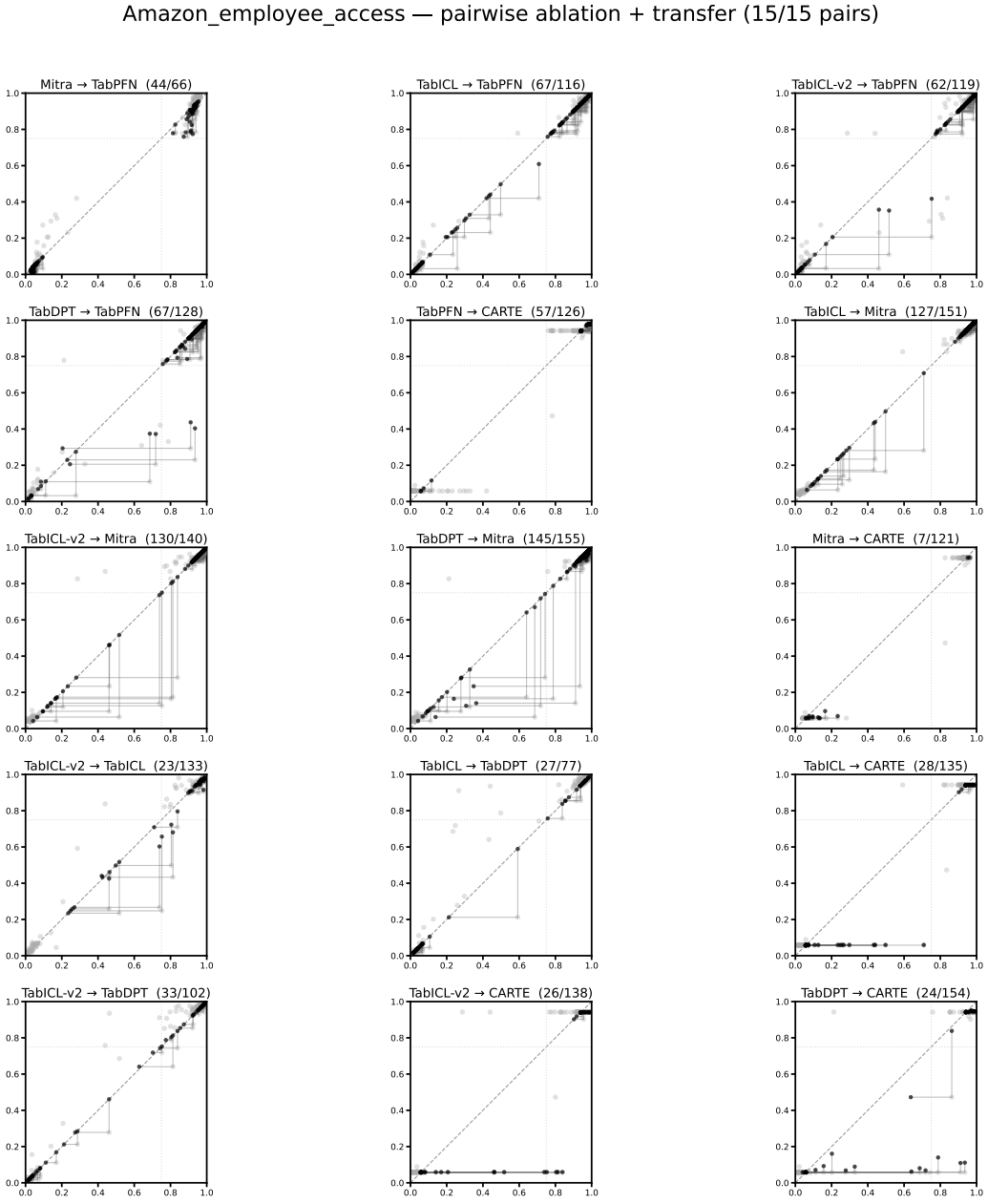


Figure 24: Amazon Employee Access

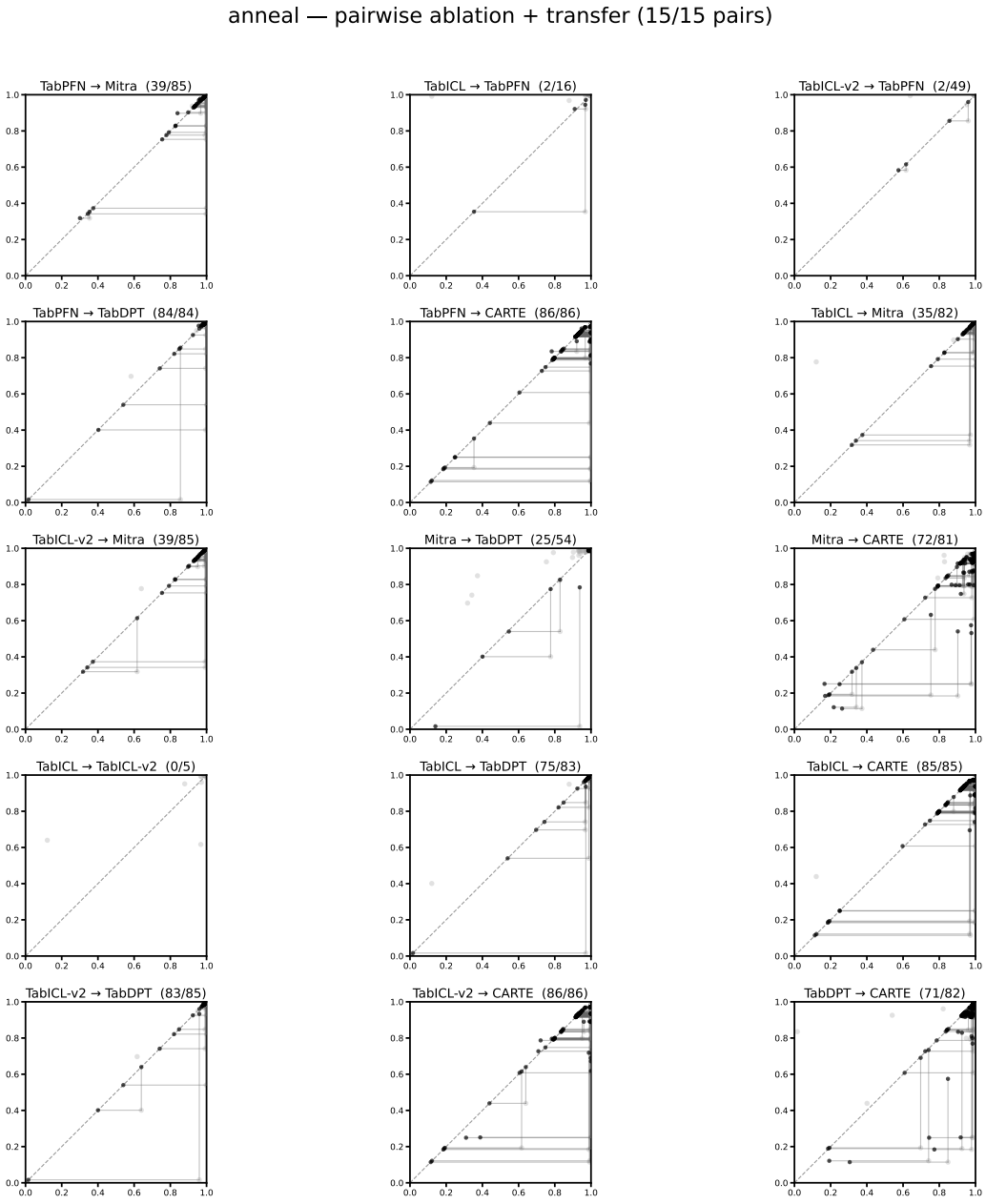


Figure 25: Anneal

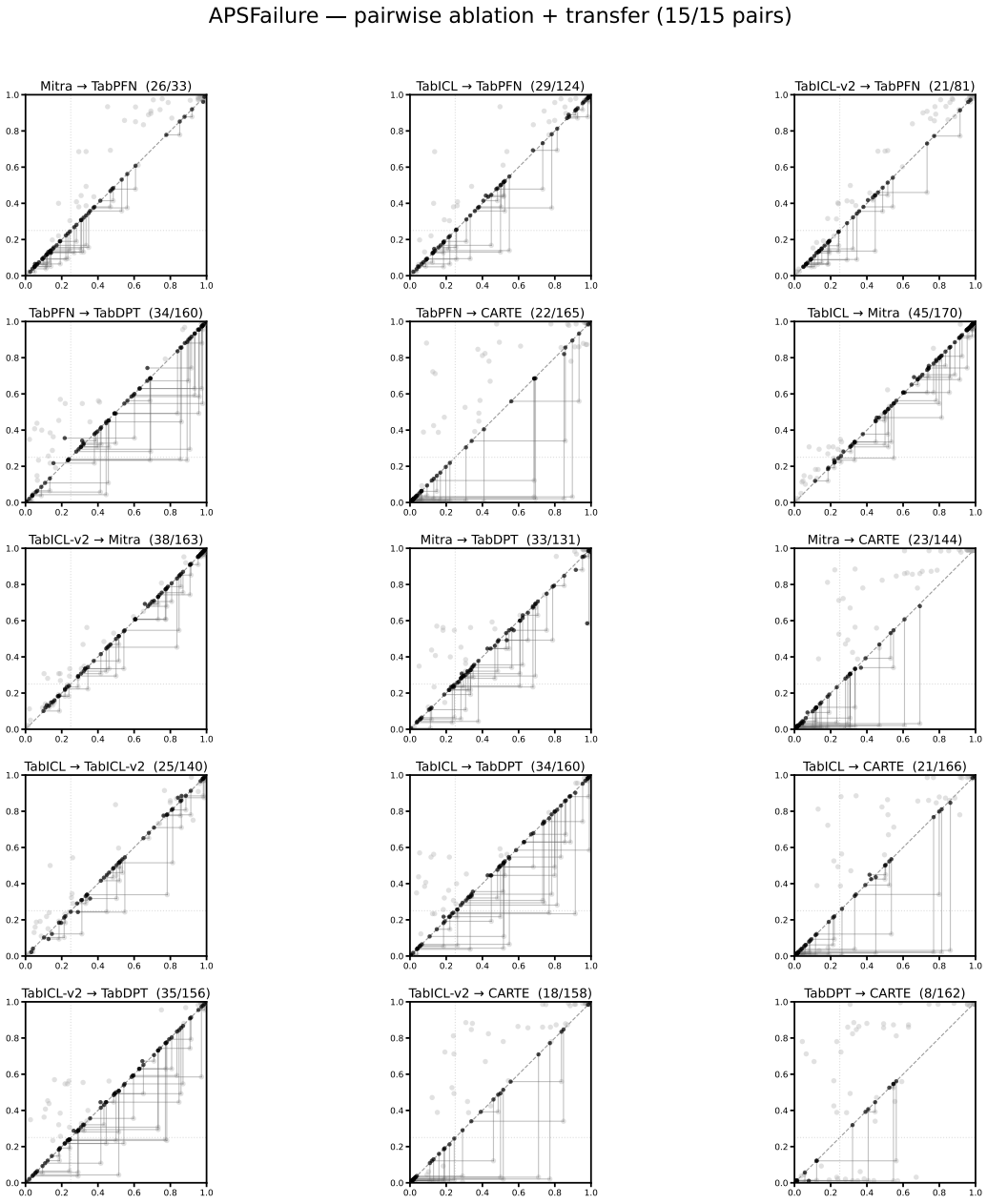


Figure 26: APS Failure

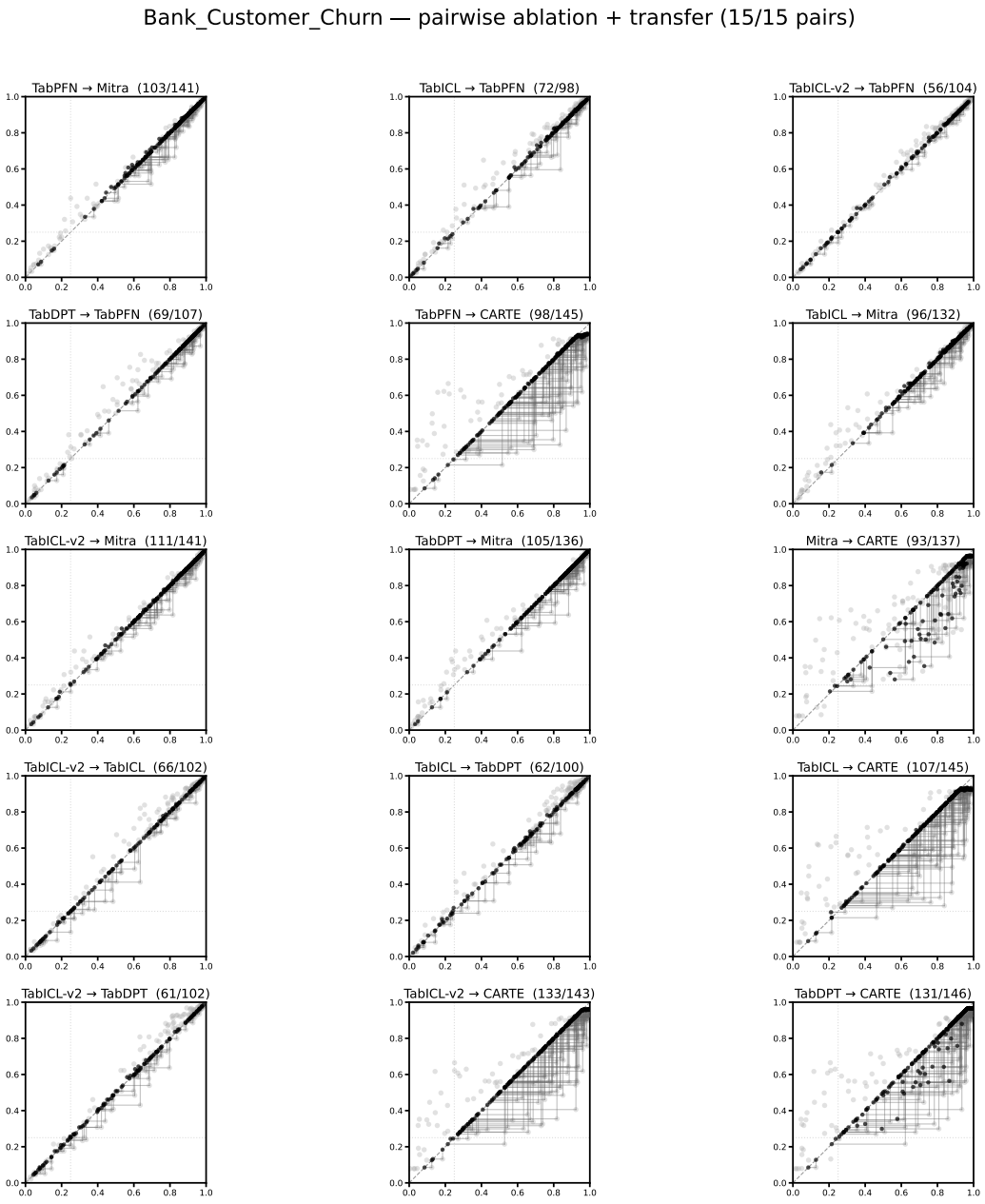


Figure 27: Bank Customer Churn

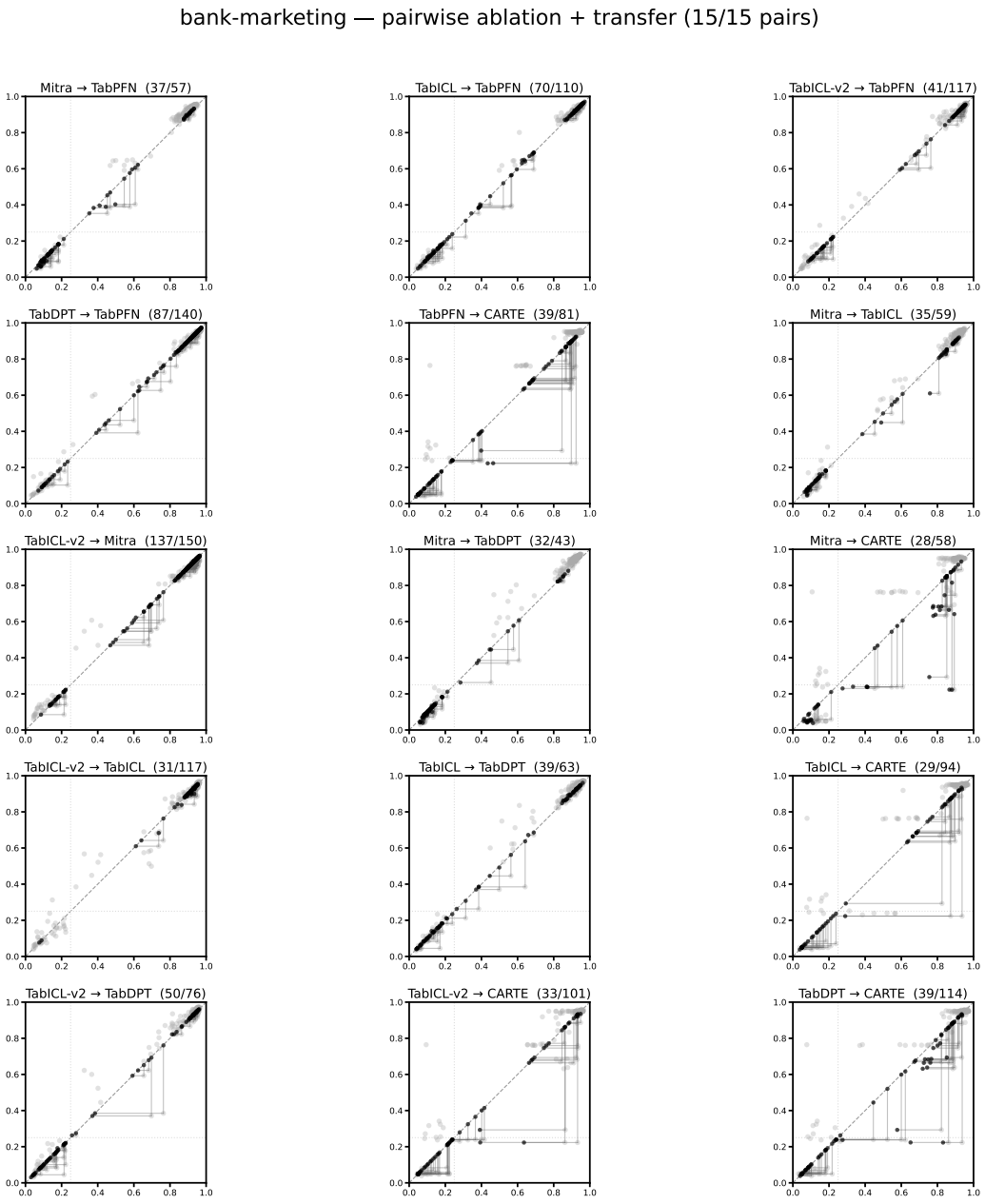


Figure 28: Bank Marketing

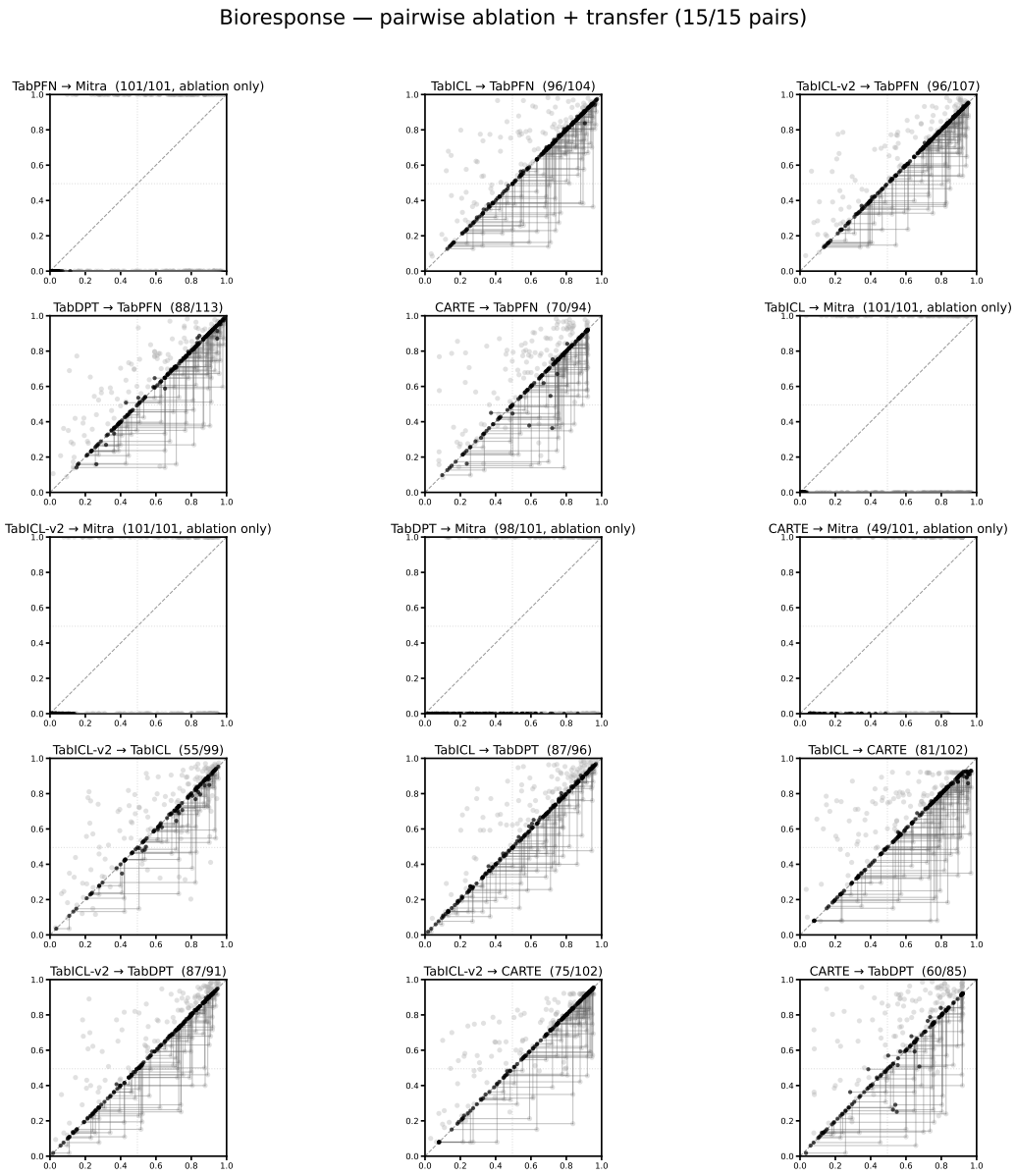


Figure 29: Bioresponse

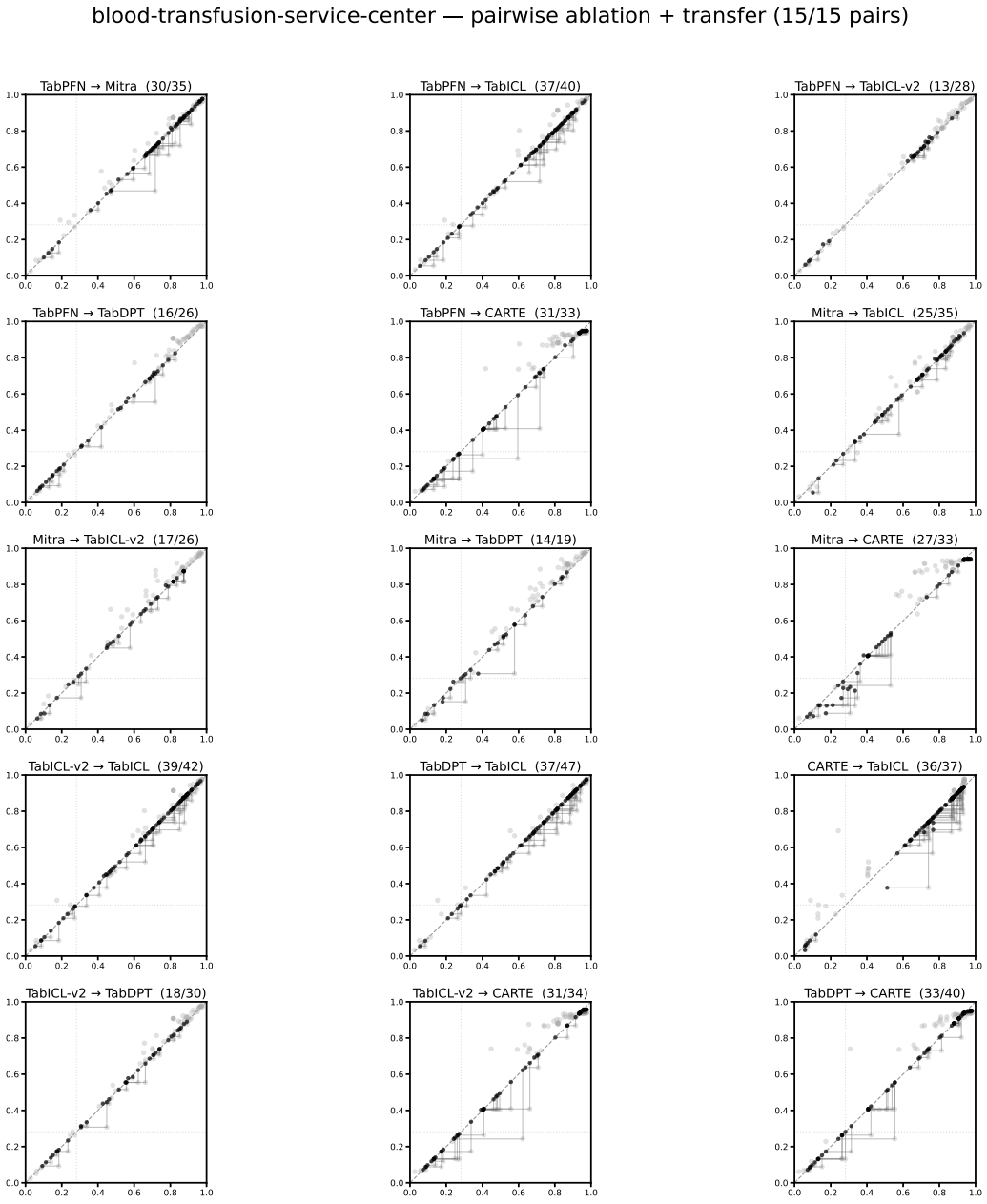


Figure 30: Blood Transfusion Service Center

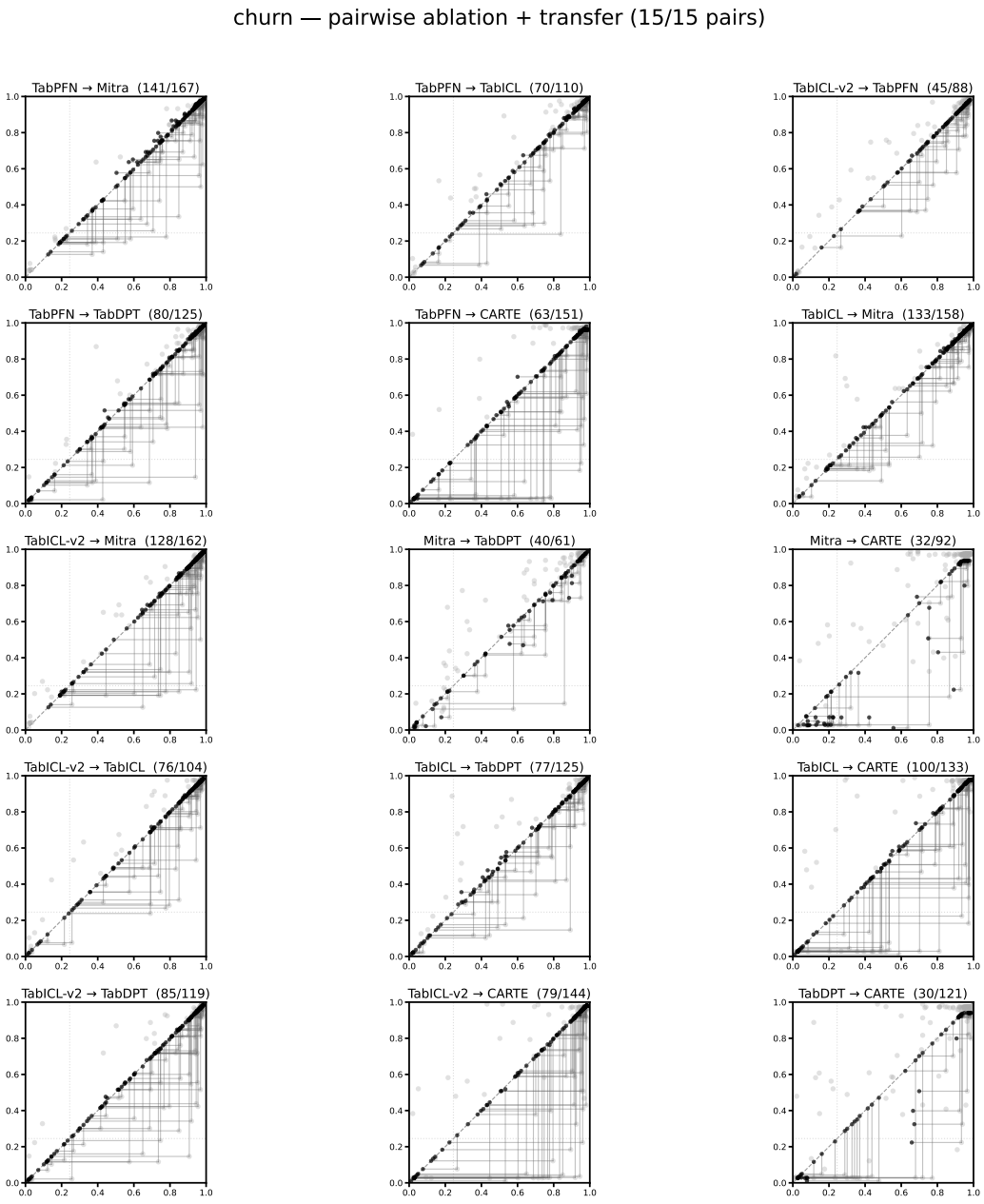


Figure 31: Churn

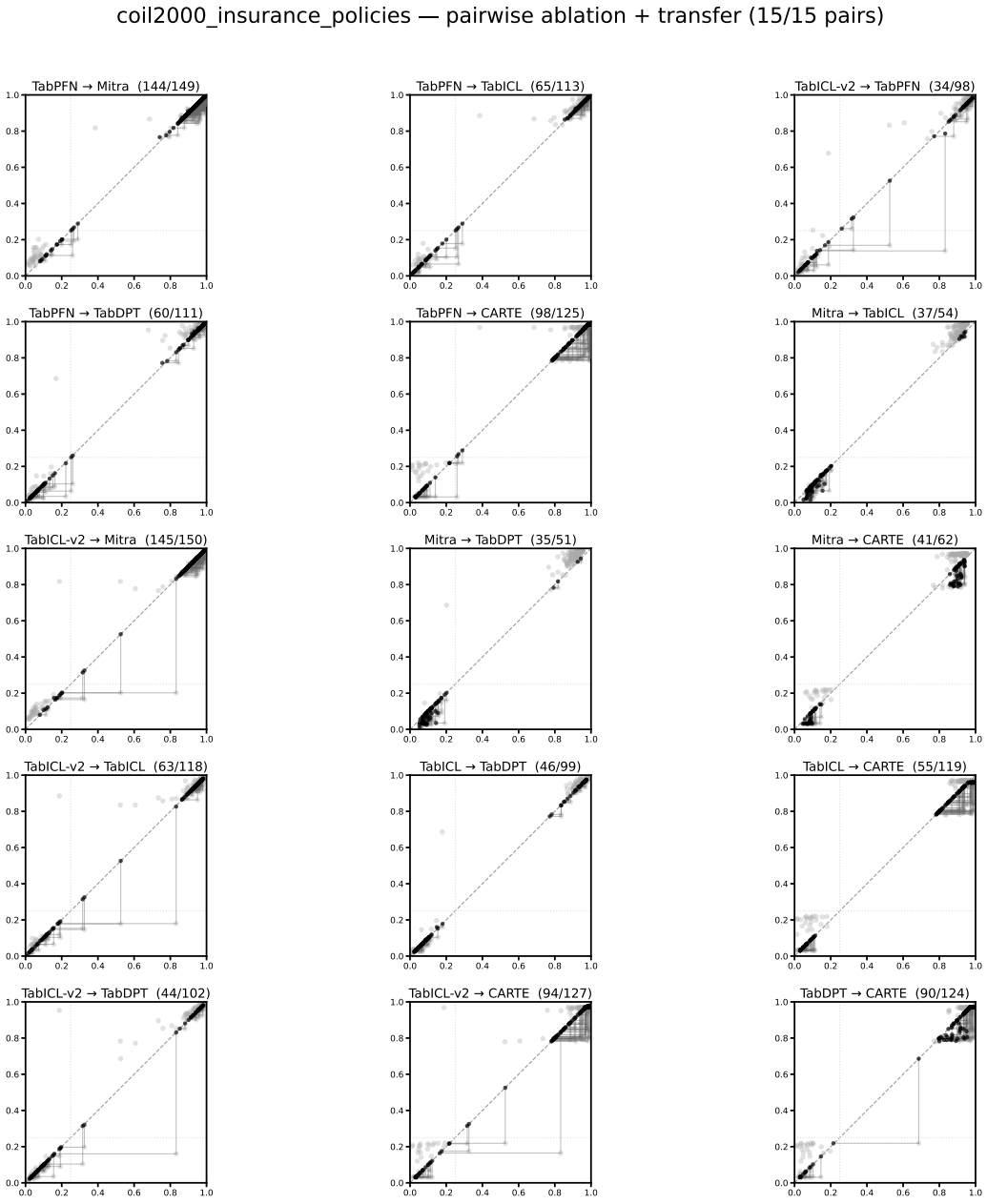


Figure 32: COIL2000 Insurance Policies

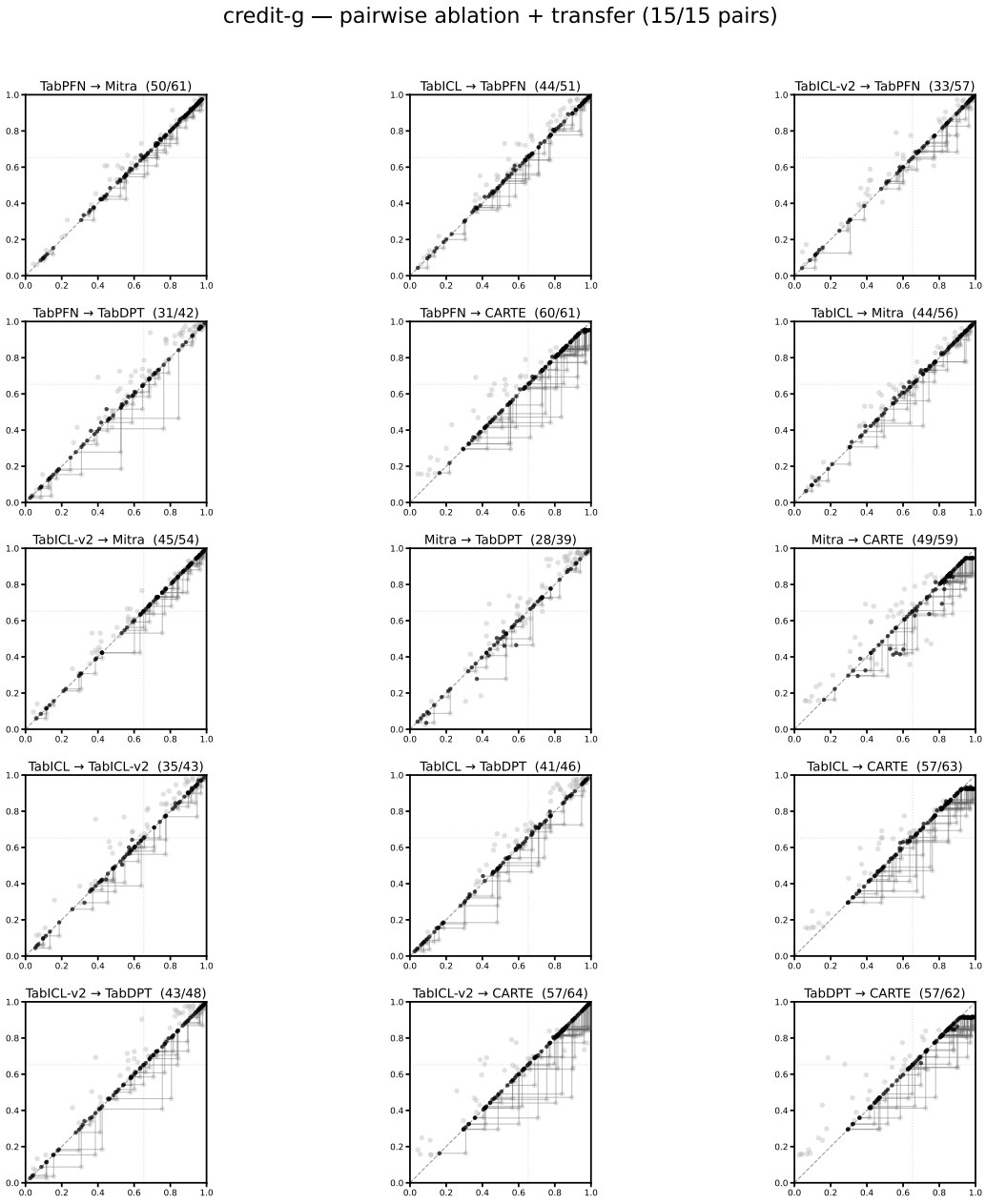


Figure 33: Credit-G

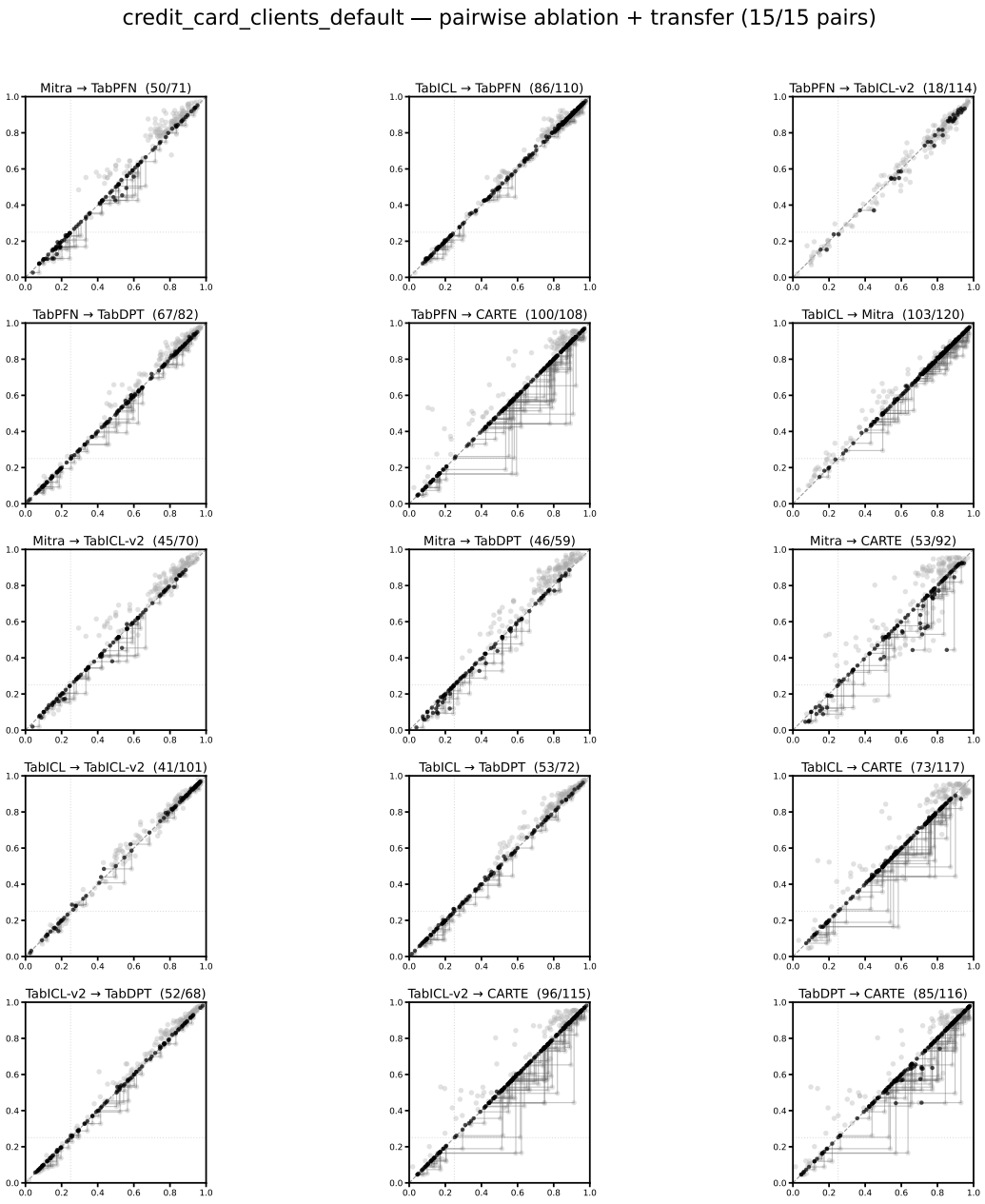


Figure 34: Credit Card Clients Default

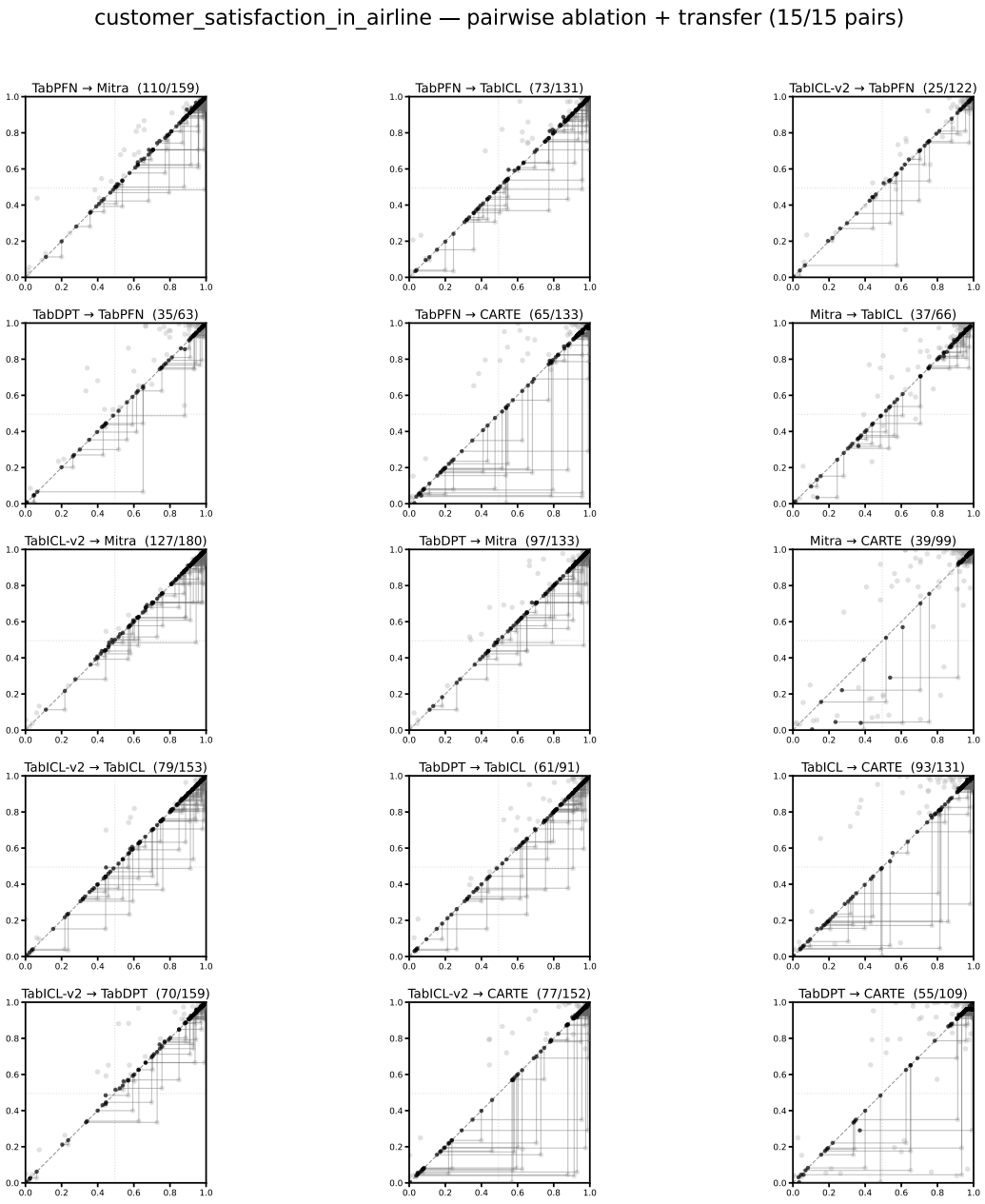


Figure 35: Customer Satisfaction in Airline

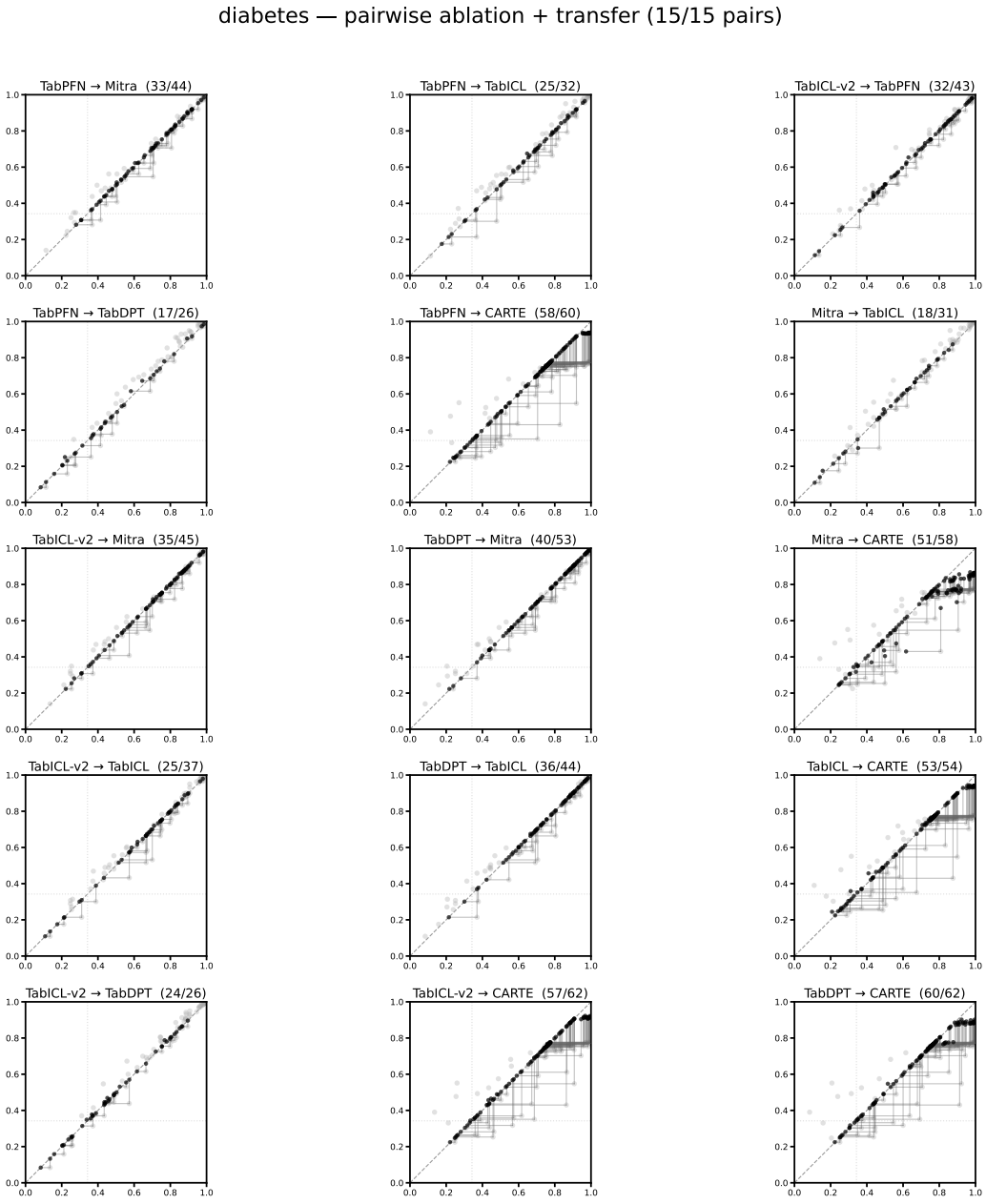


Figure 36: Diabetes

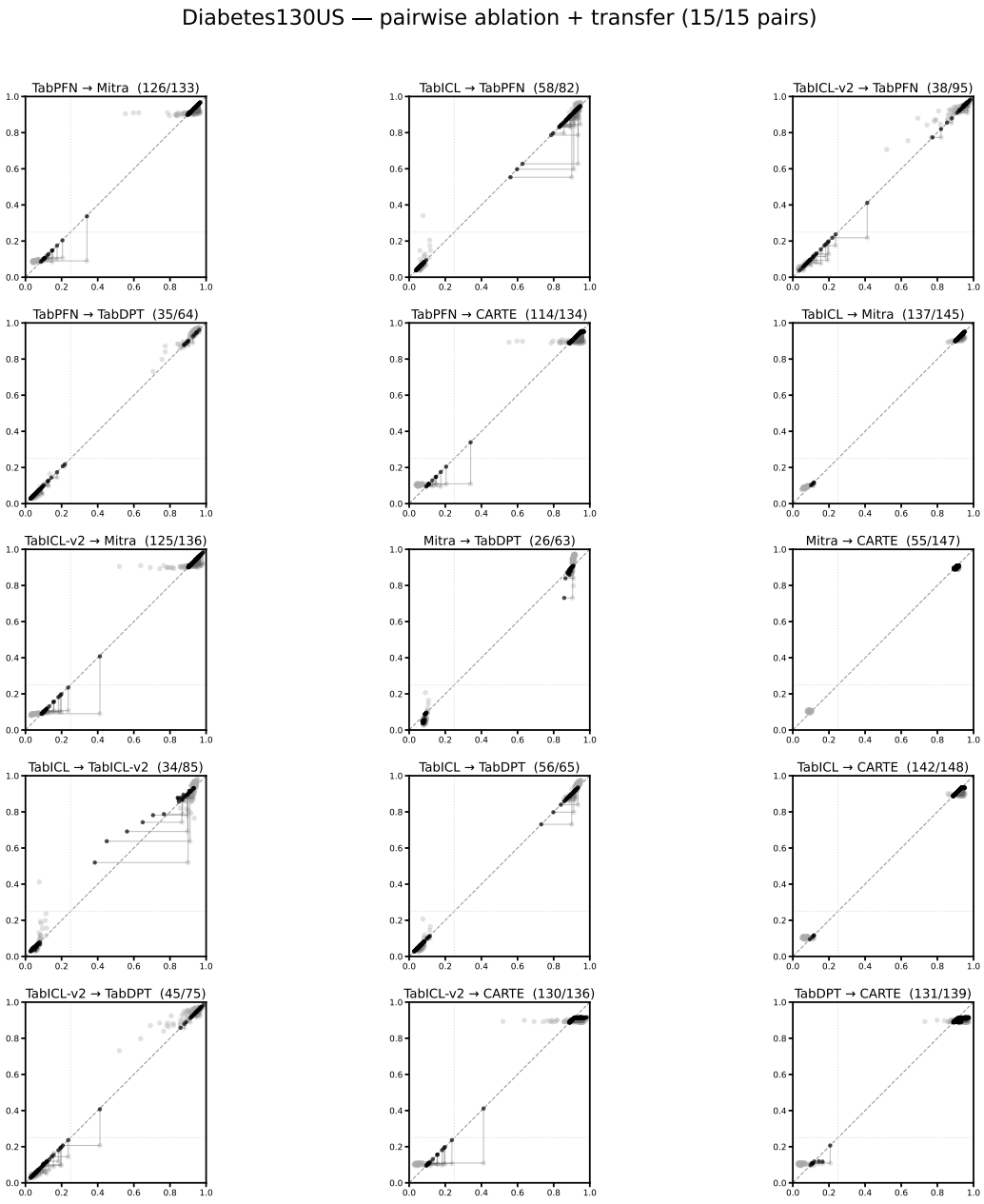


Figure 37: Diabetes130US

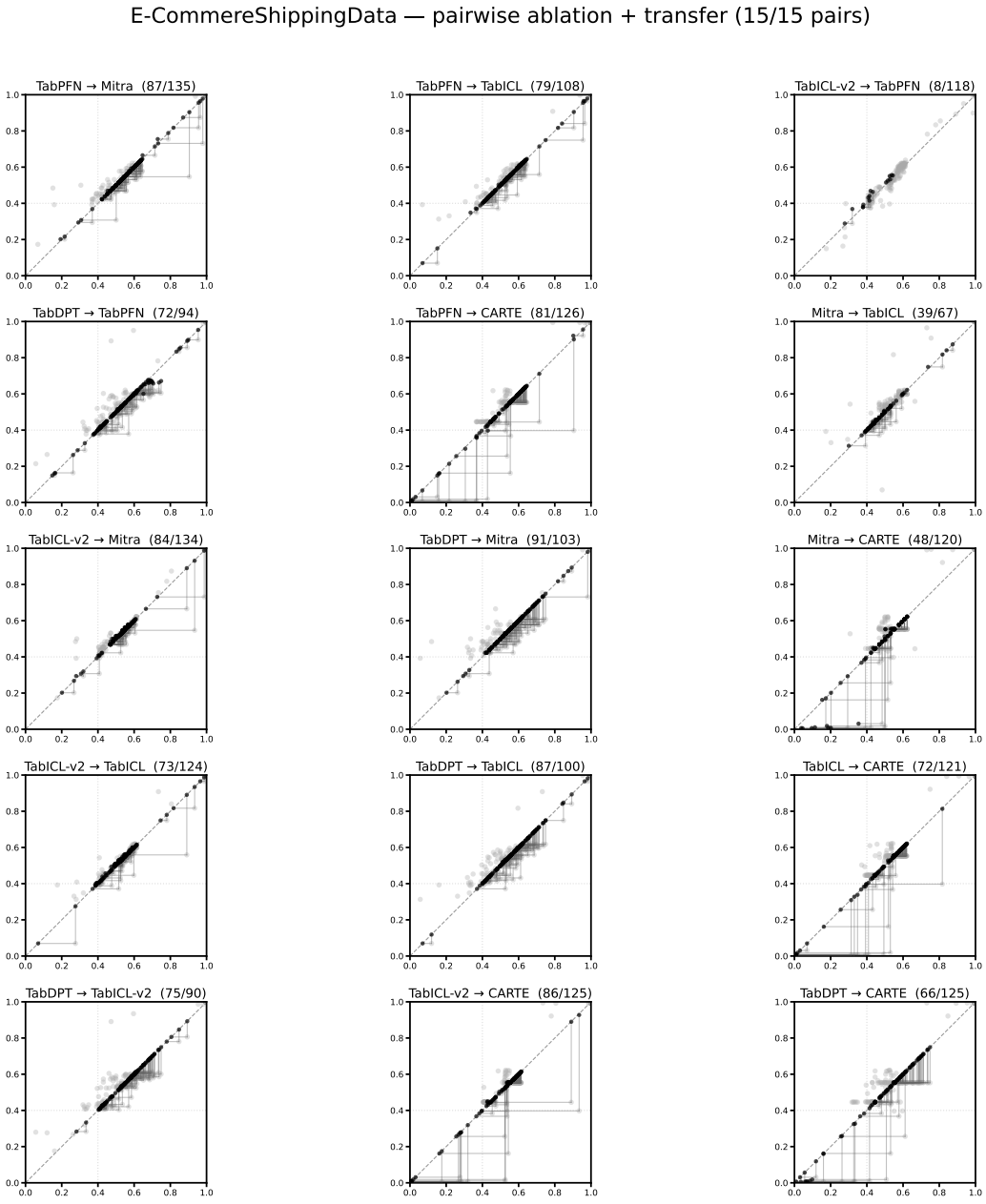


Figure 38: E-Commerce Shipping Data

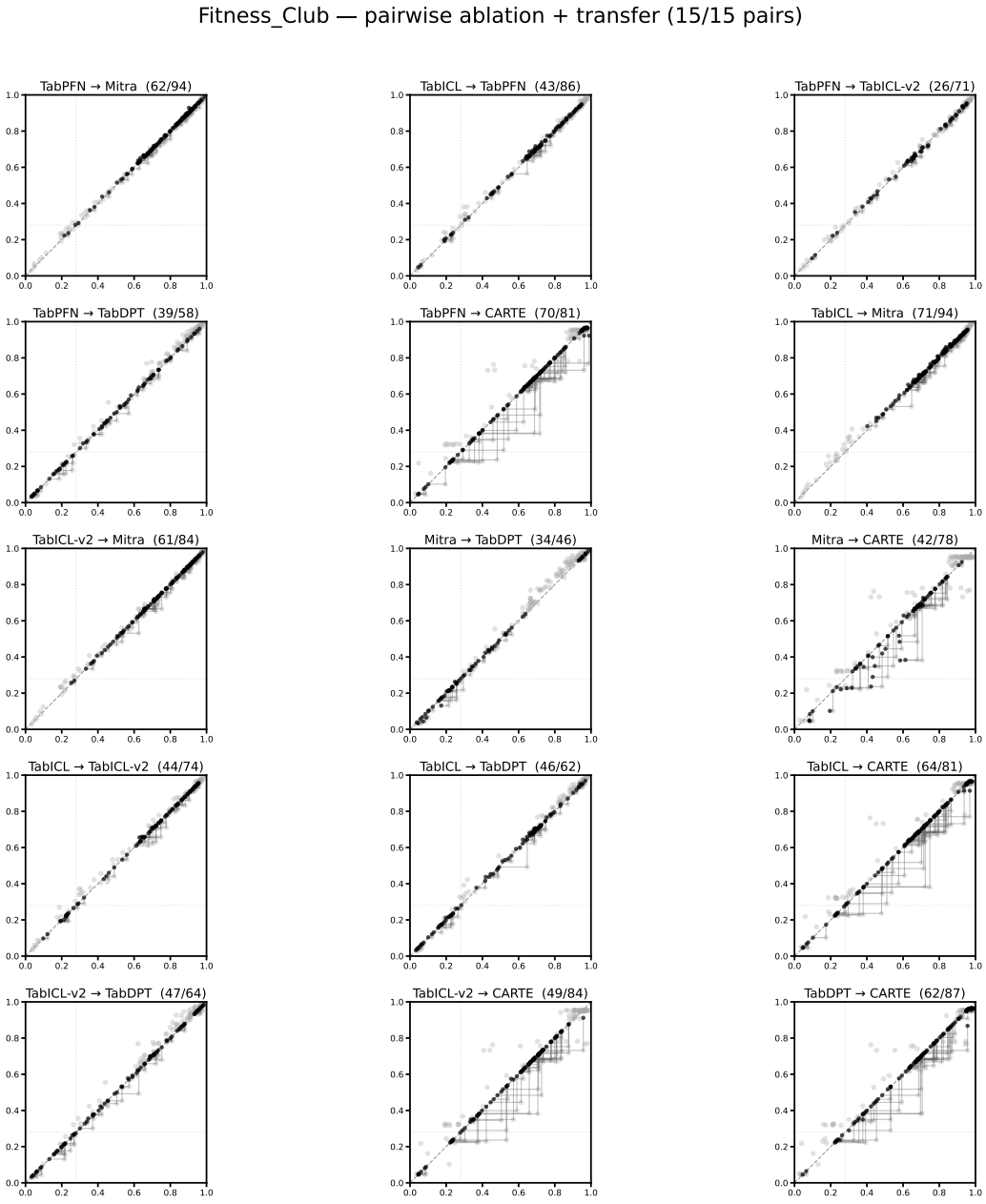


Figure 39: Fitness Club

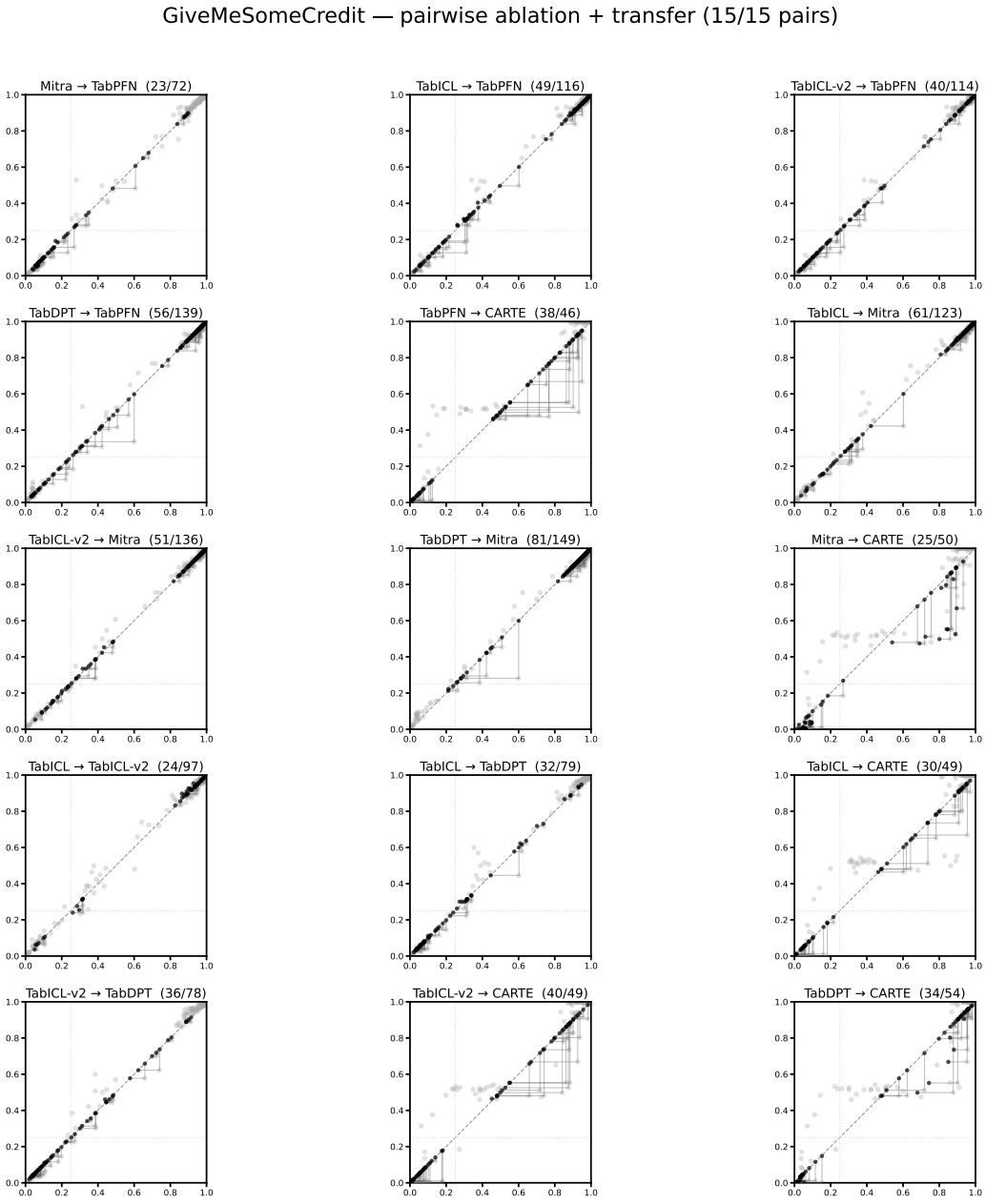


Figure 40: Give Me Some Credit

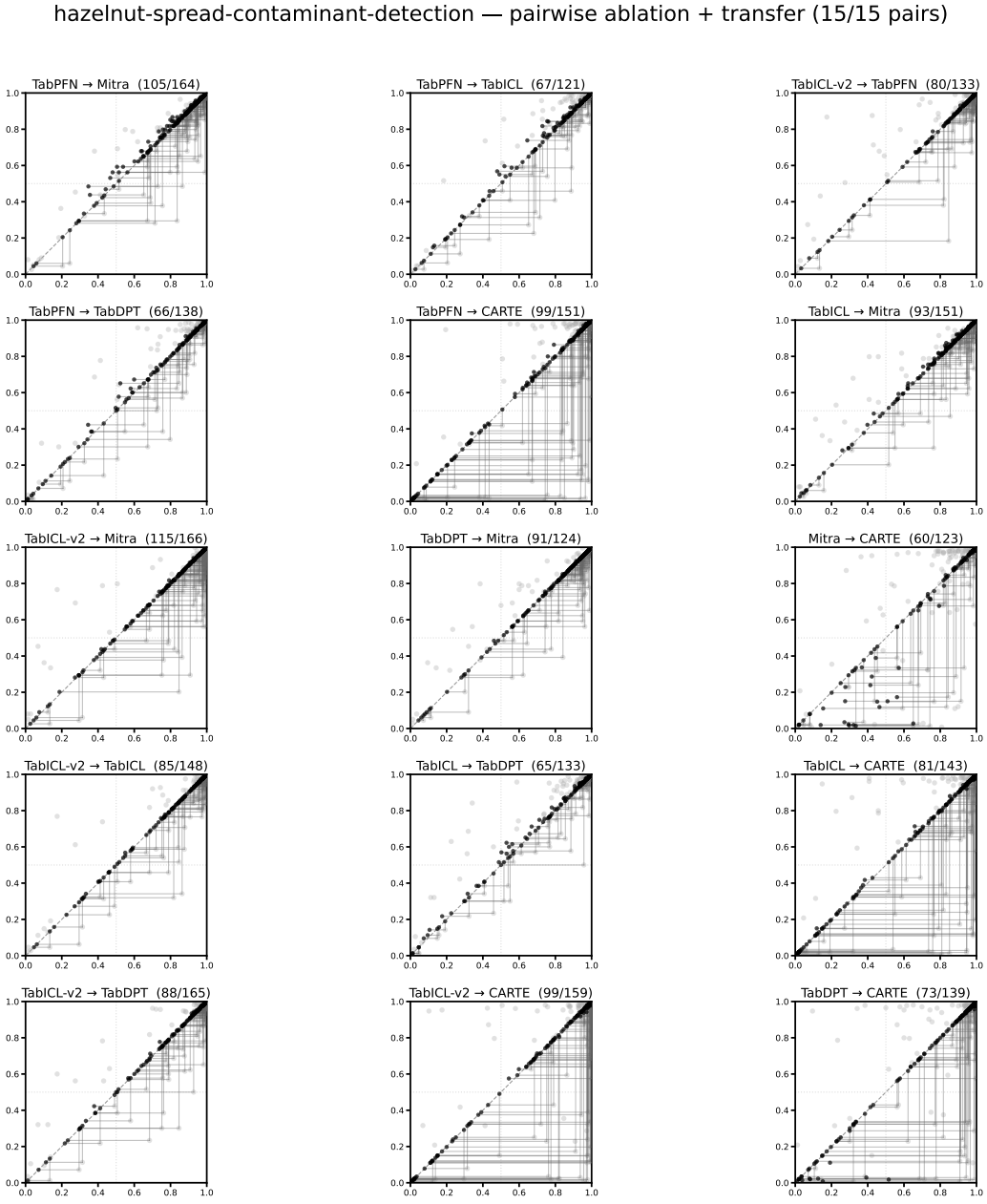


Figure 41: Hazelnut Spread Contaminant Detection

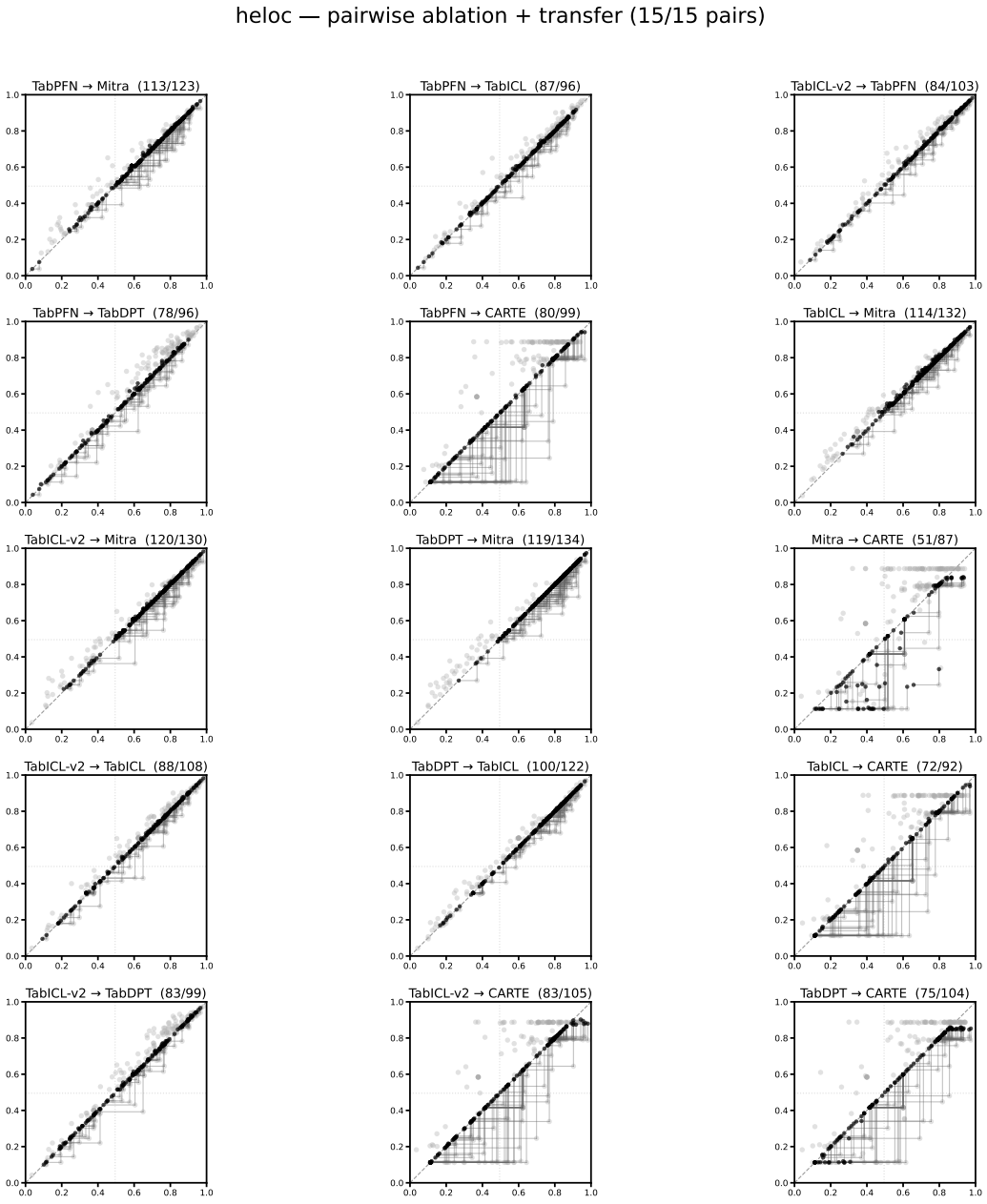


Figure 42: HELOC

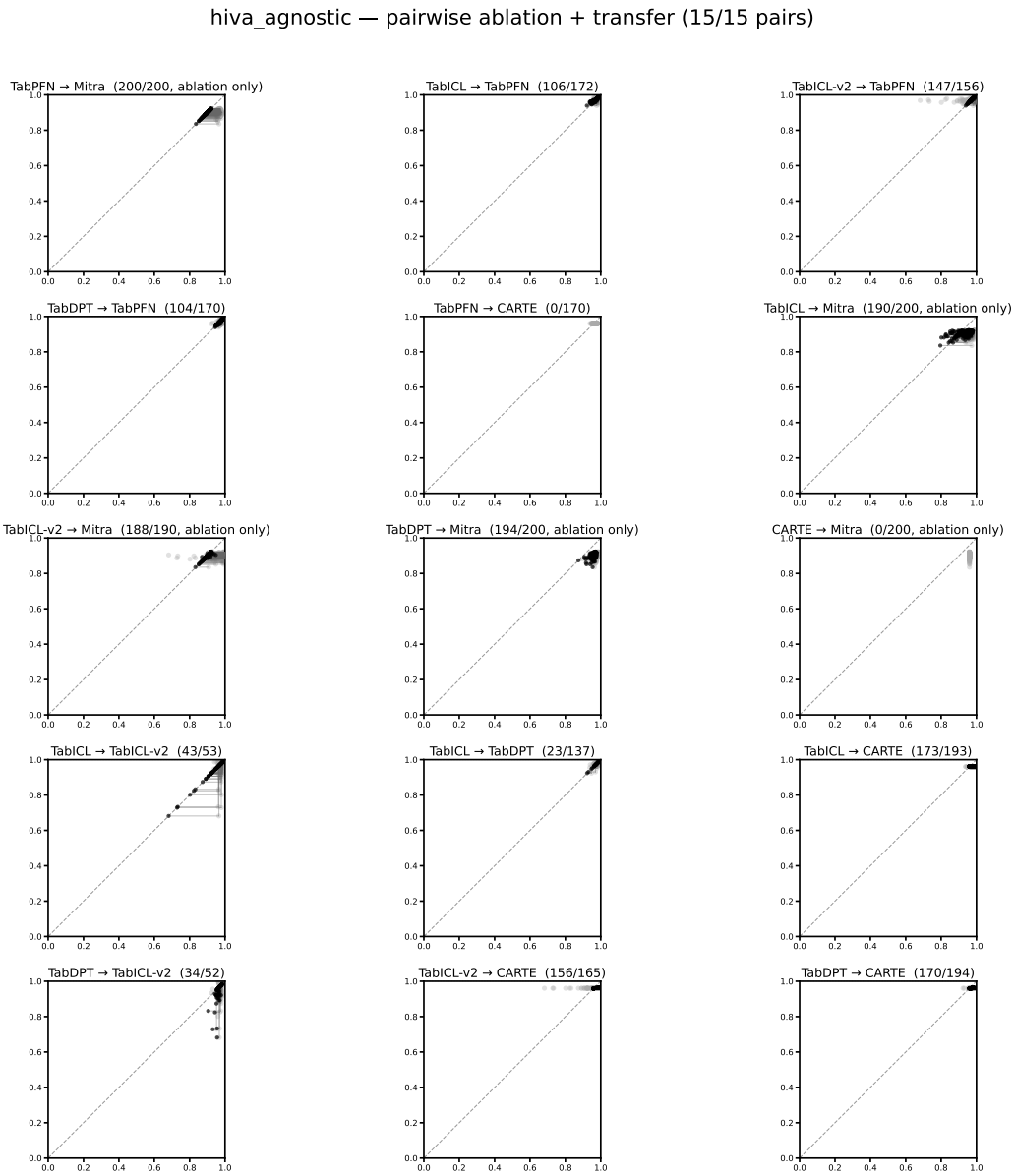


Figure 43: HIVA Agnostic

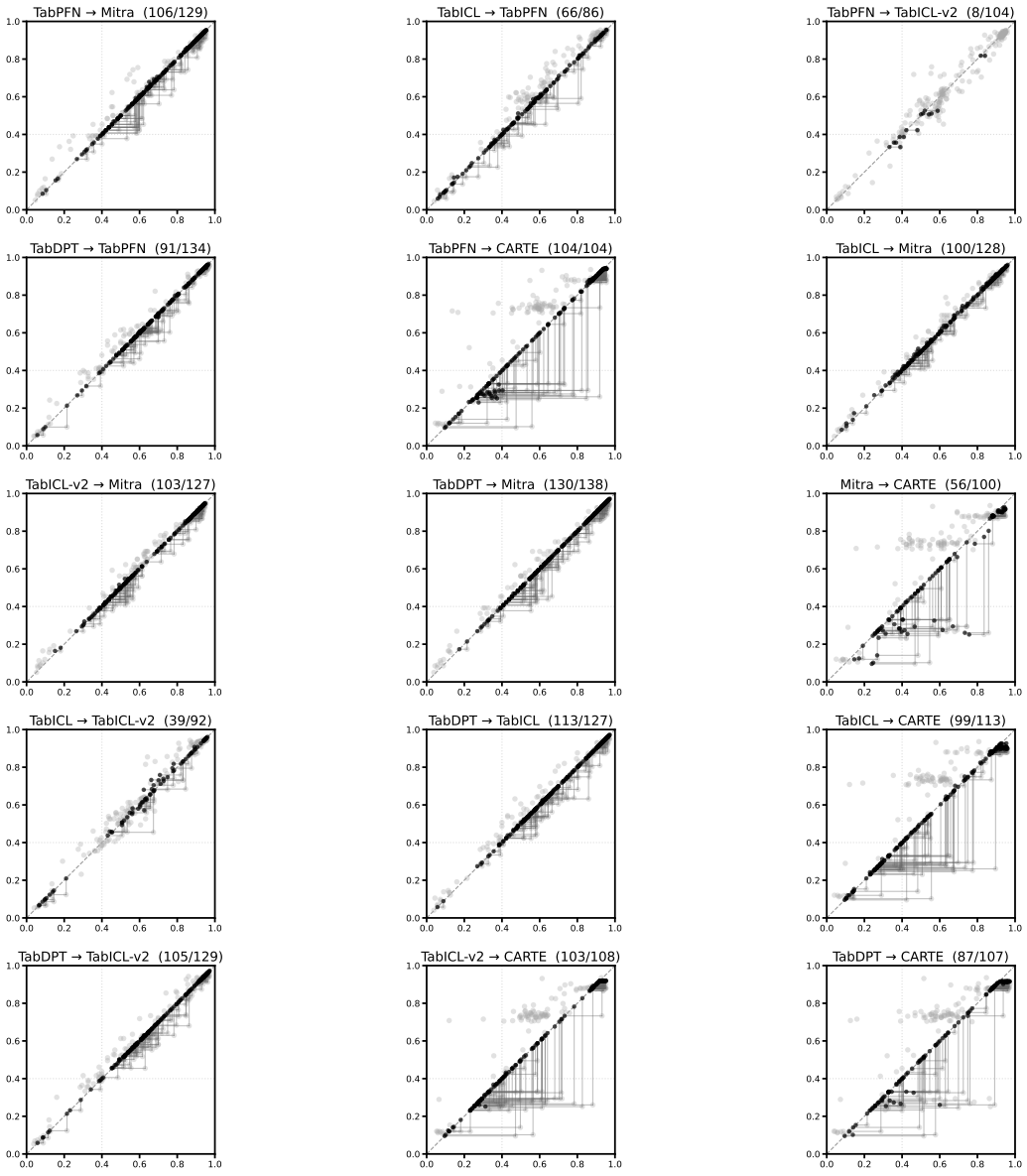


Figure 44: HR Analytics: Job Change of Data Scientists

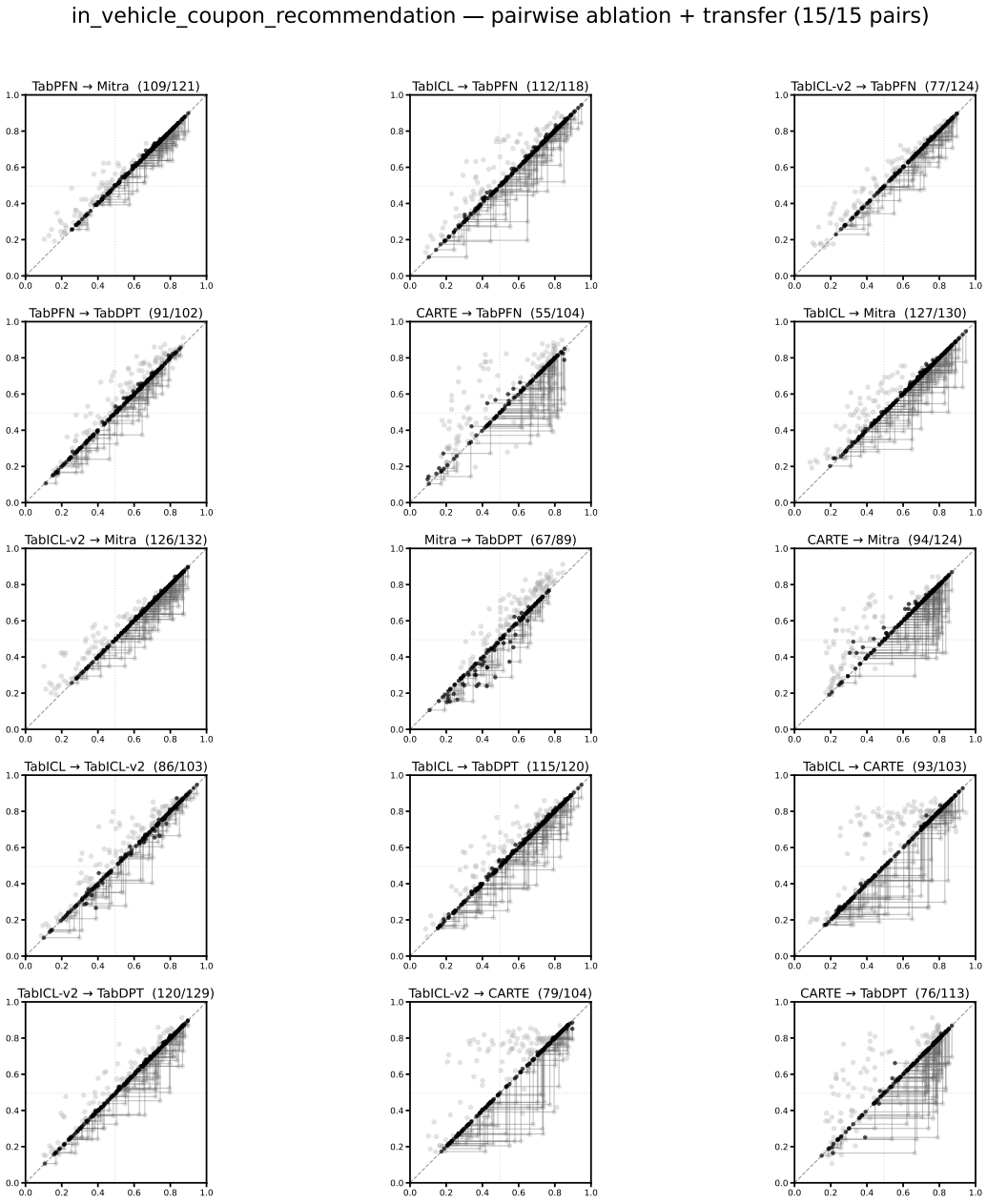


Figure 45: In-Vehicle Coupon Recommendation

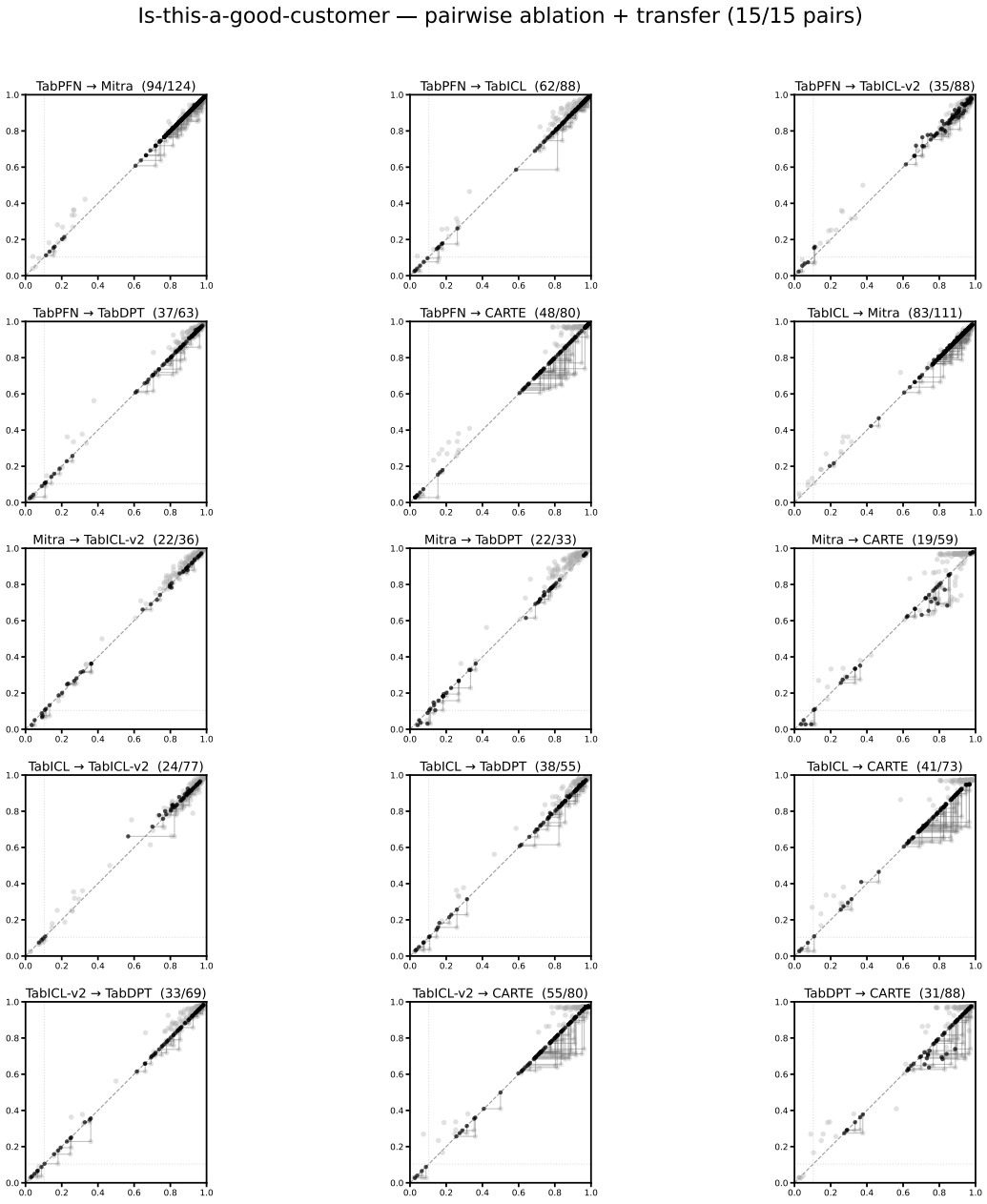


Figure 46: Is This a Good Customer?

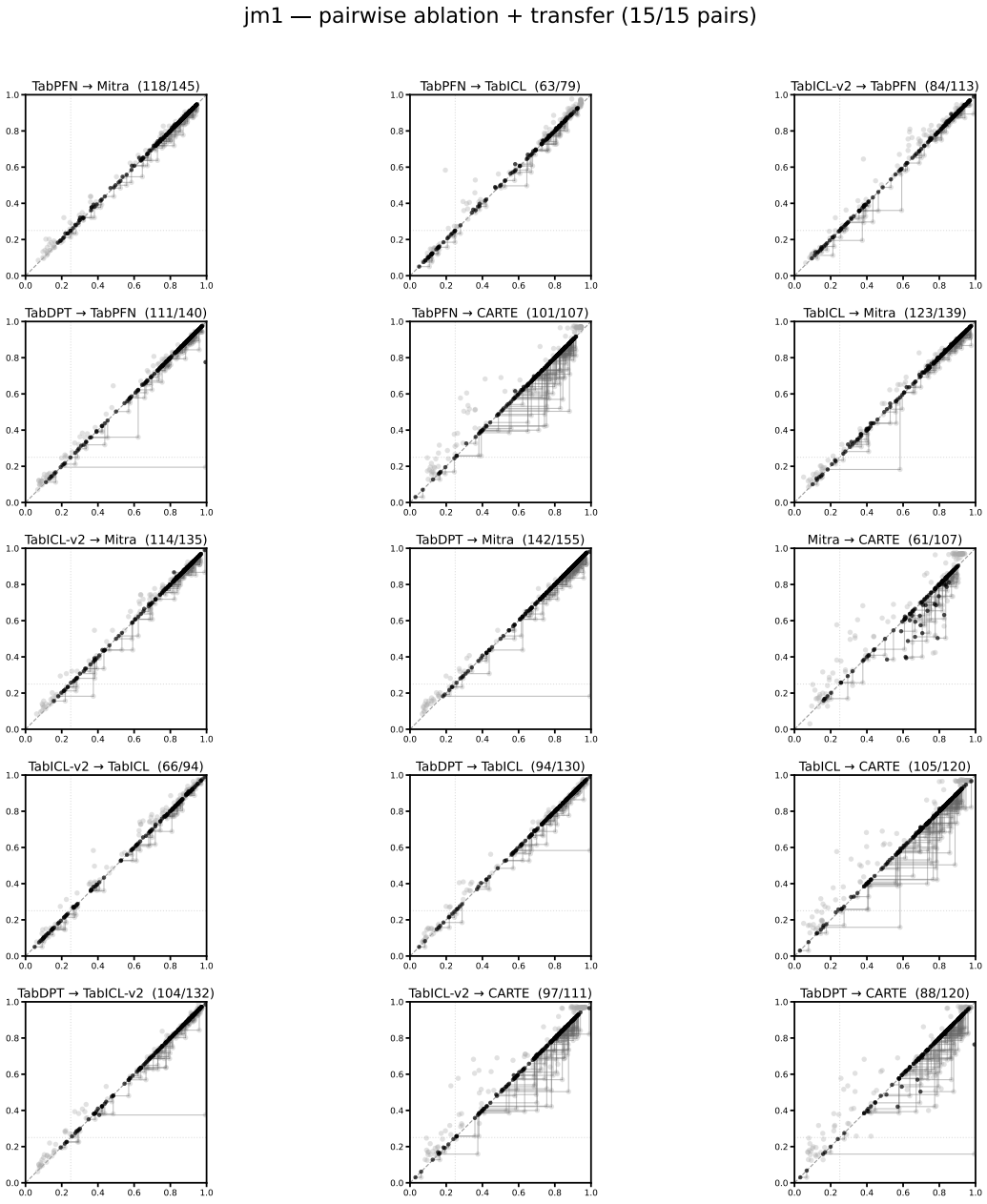


Figure 47: JM1

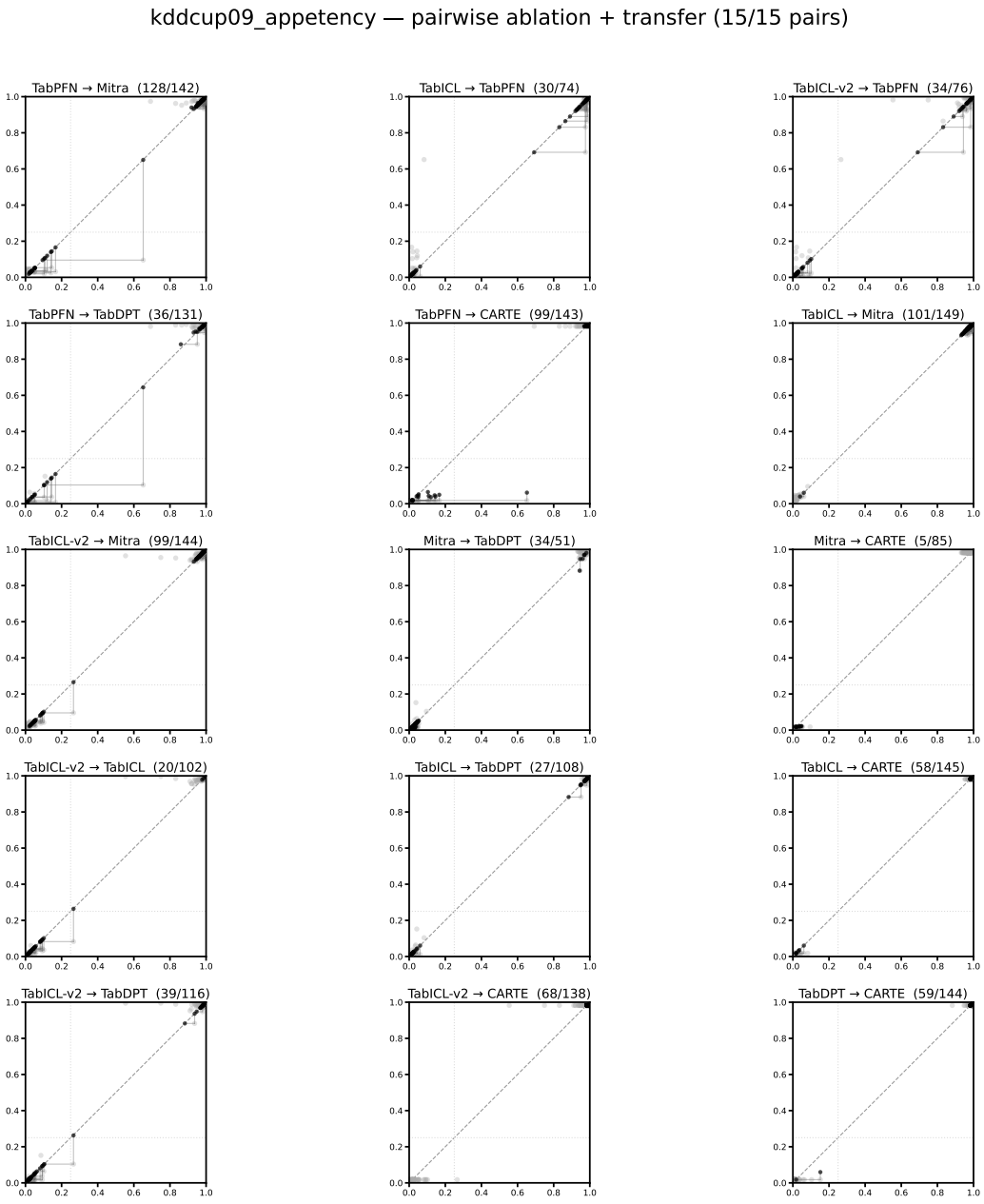


Figure 48: KDD Cup 2009 Appetency

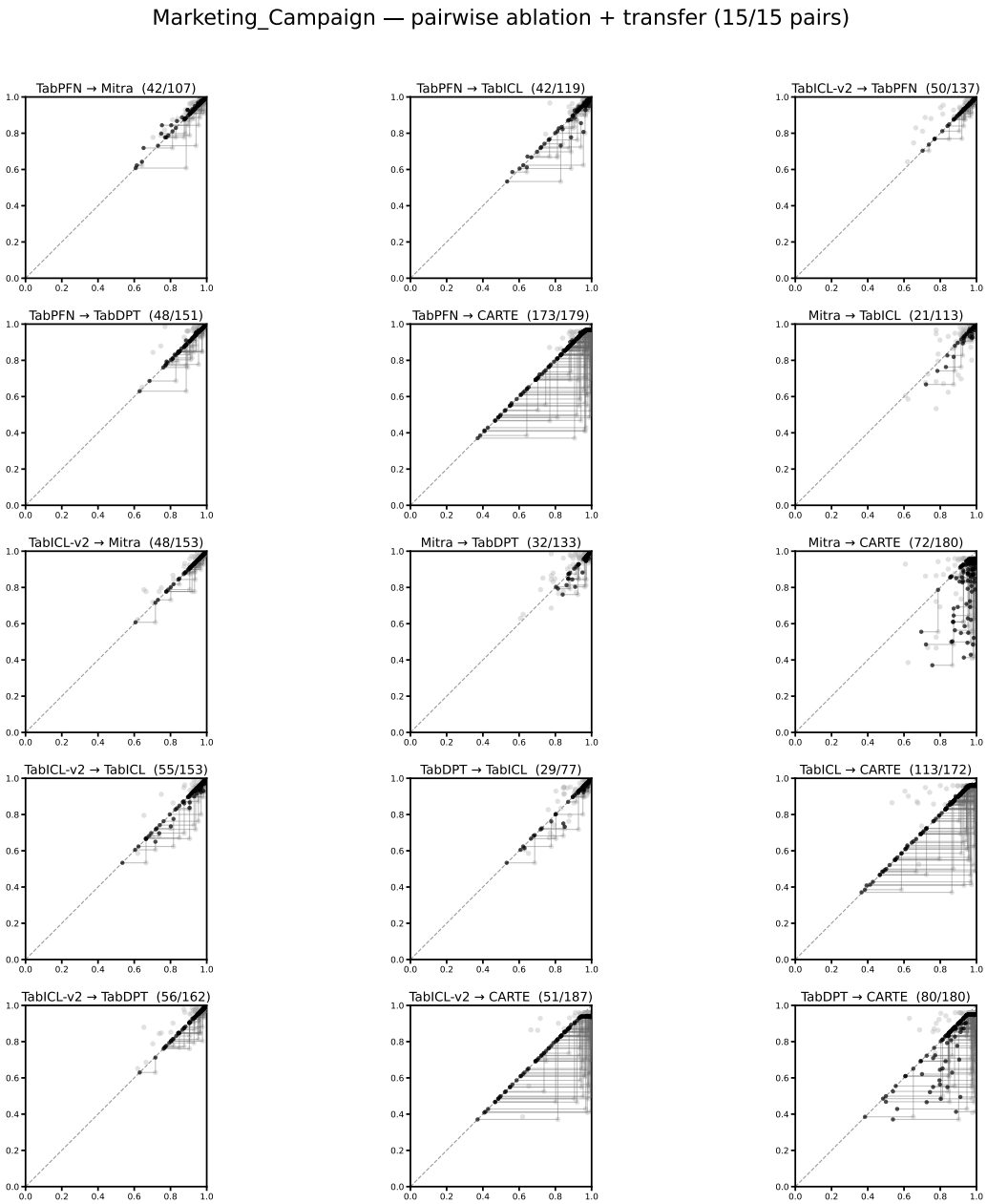


Figure 49: Marketing Campaign

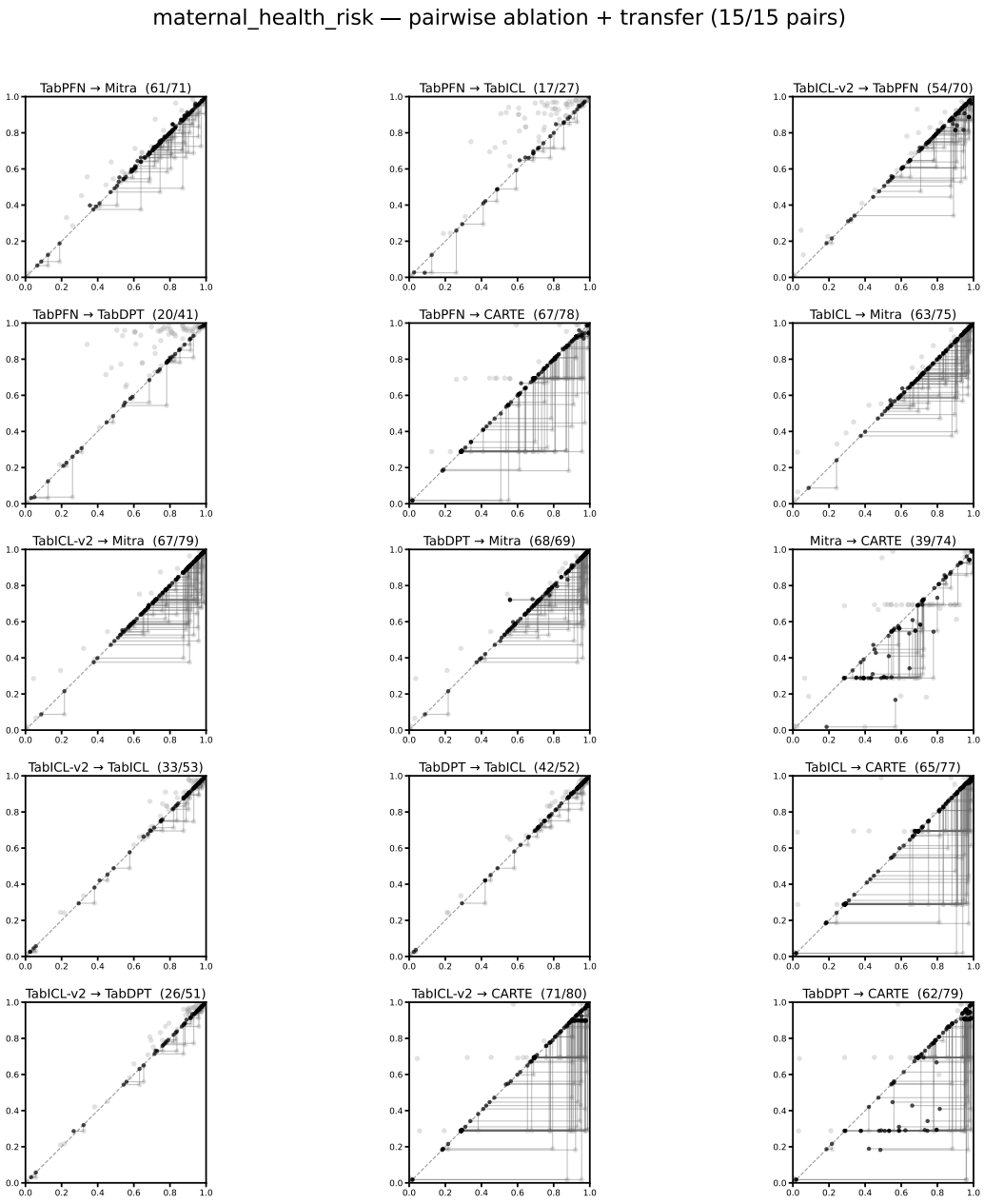


Figure 50: Maternal Health Risk

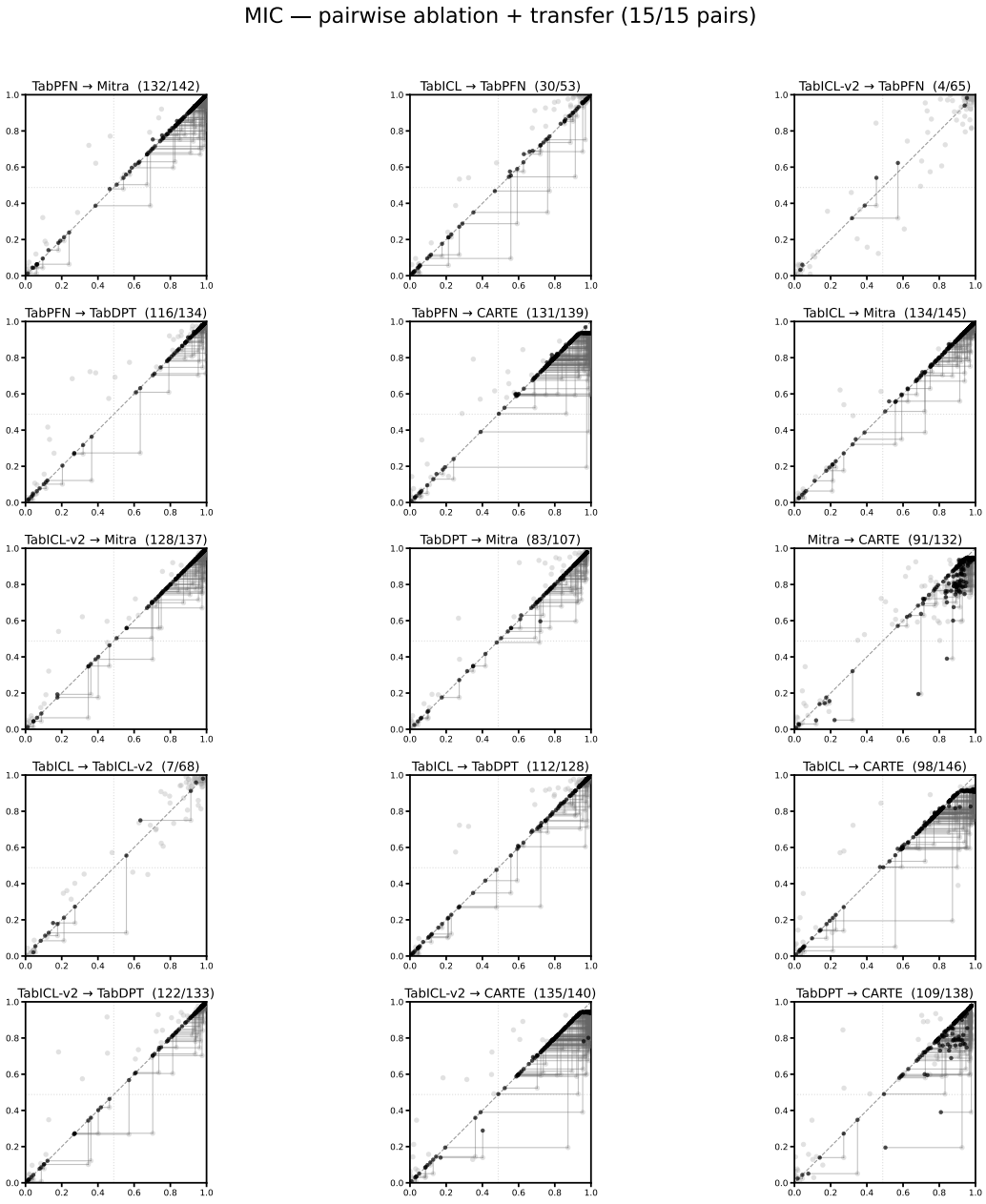


Figure 51: MIC

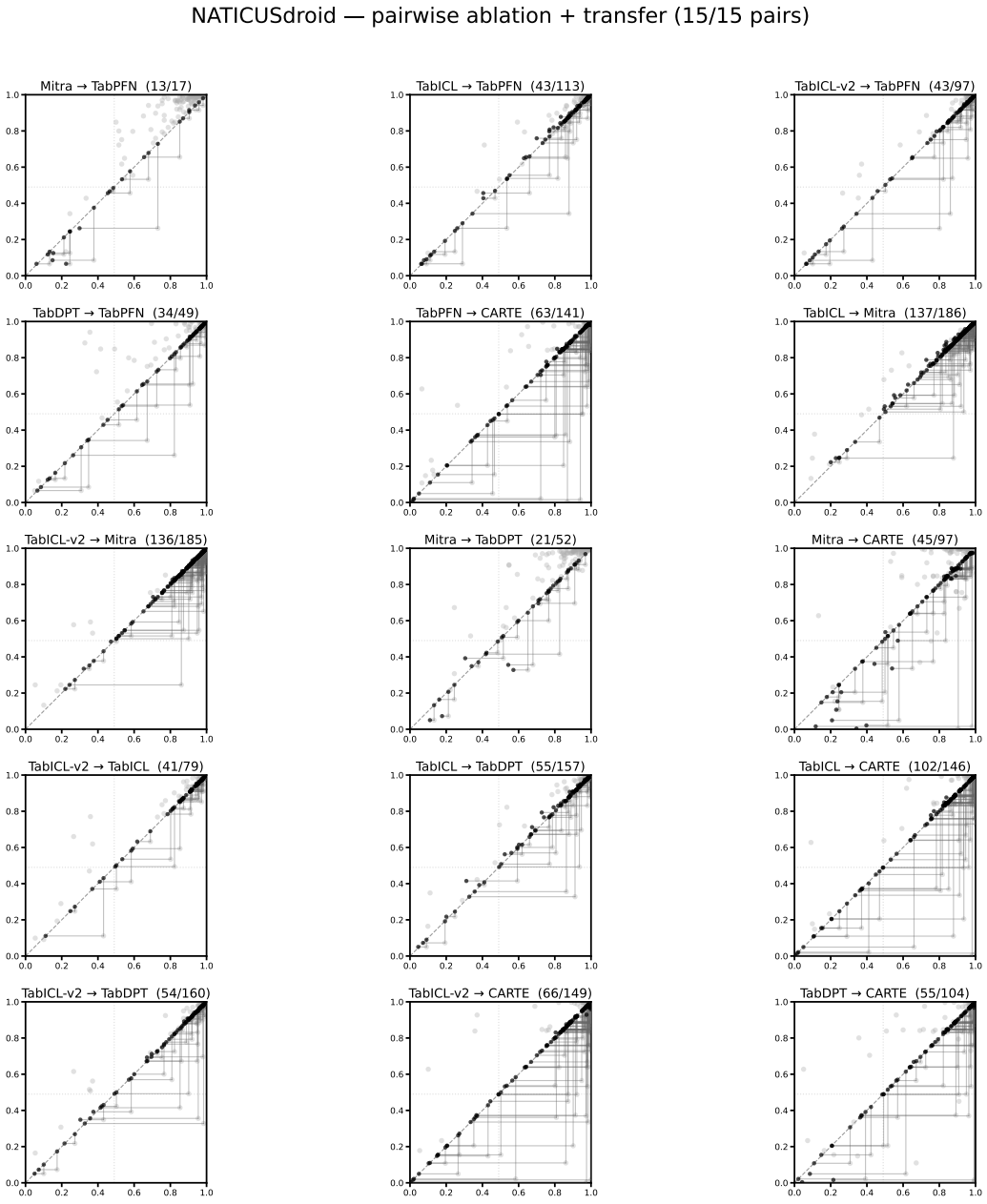


Figure 52: NATICUSdroid

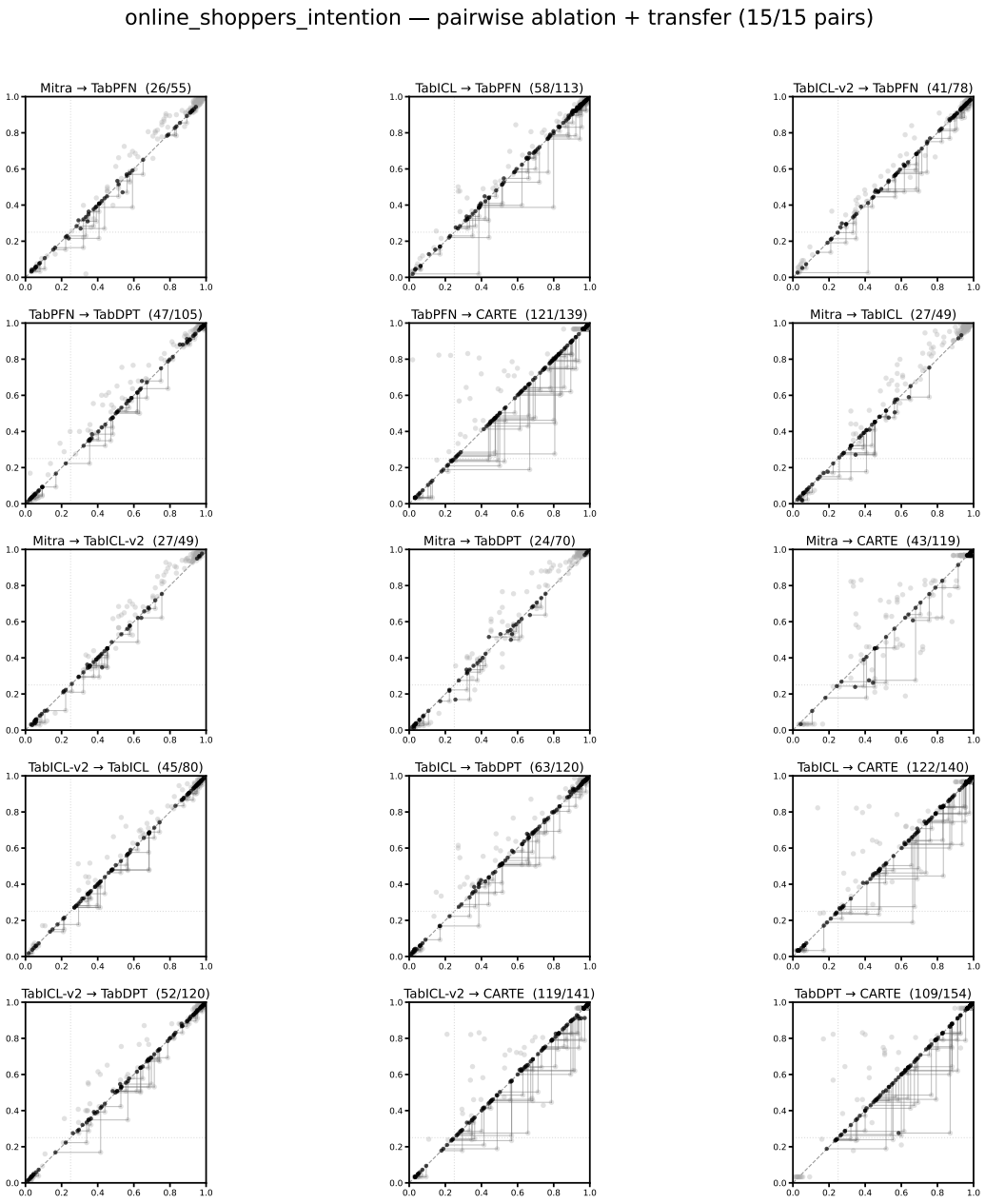


Figure 53: Online Shoppers Intention

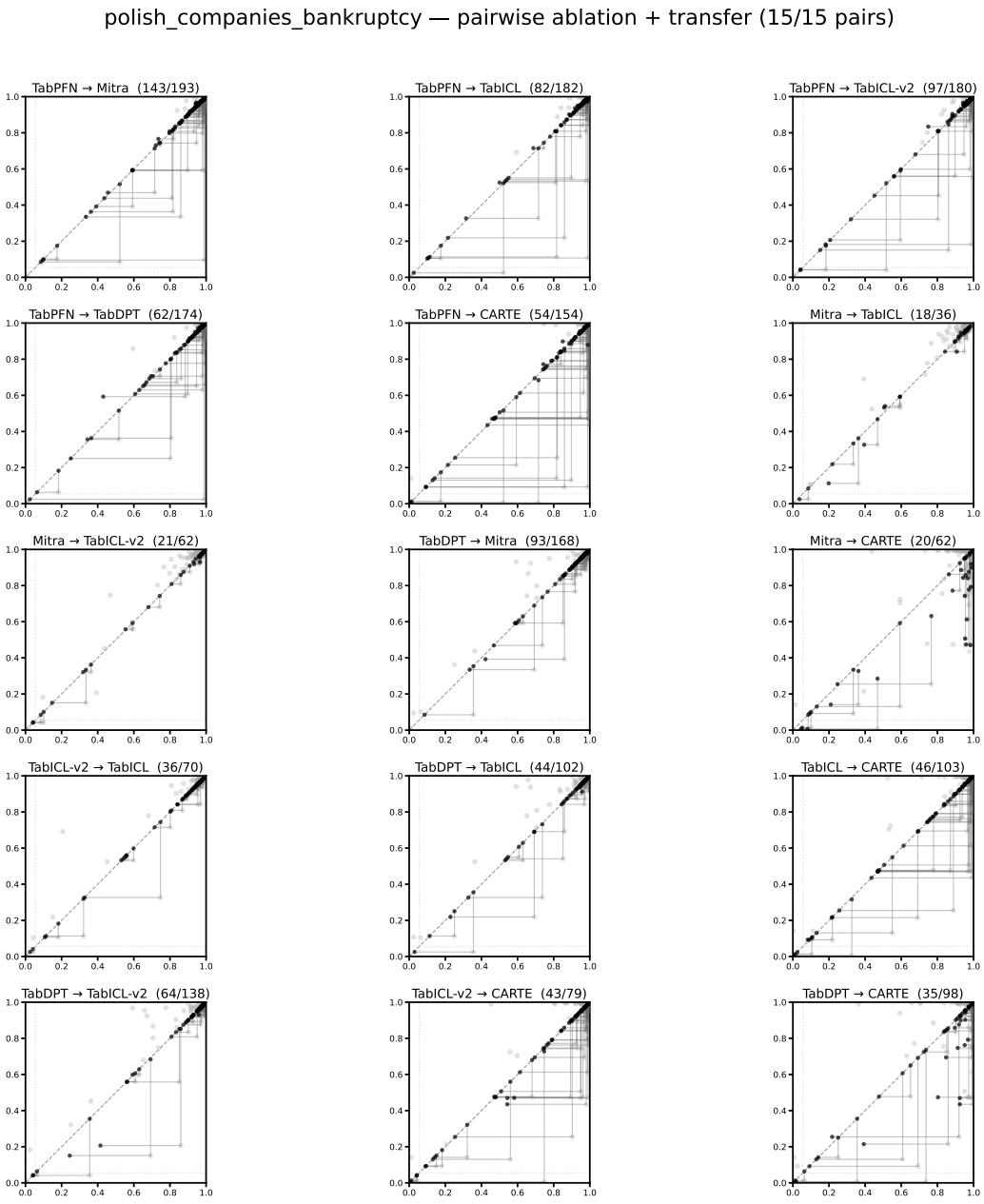


Figure 54: Polish Companies Bankruptcy

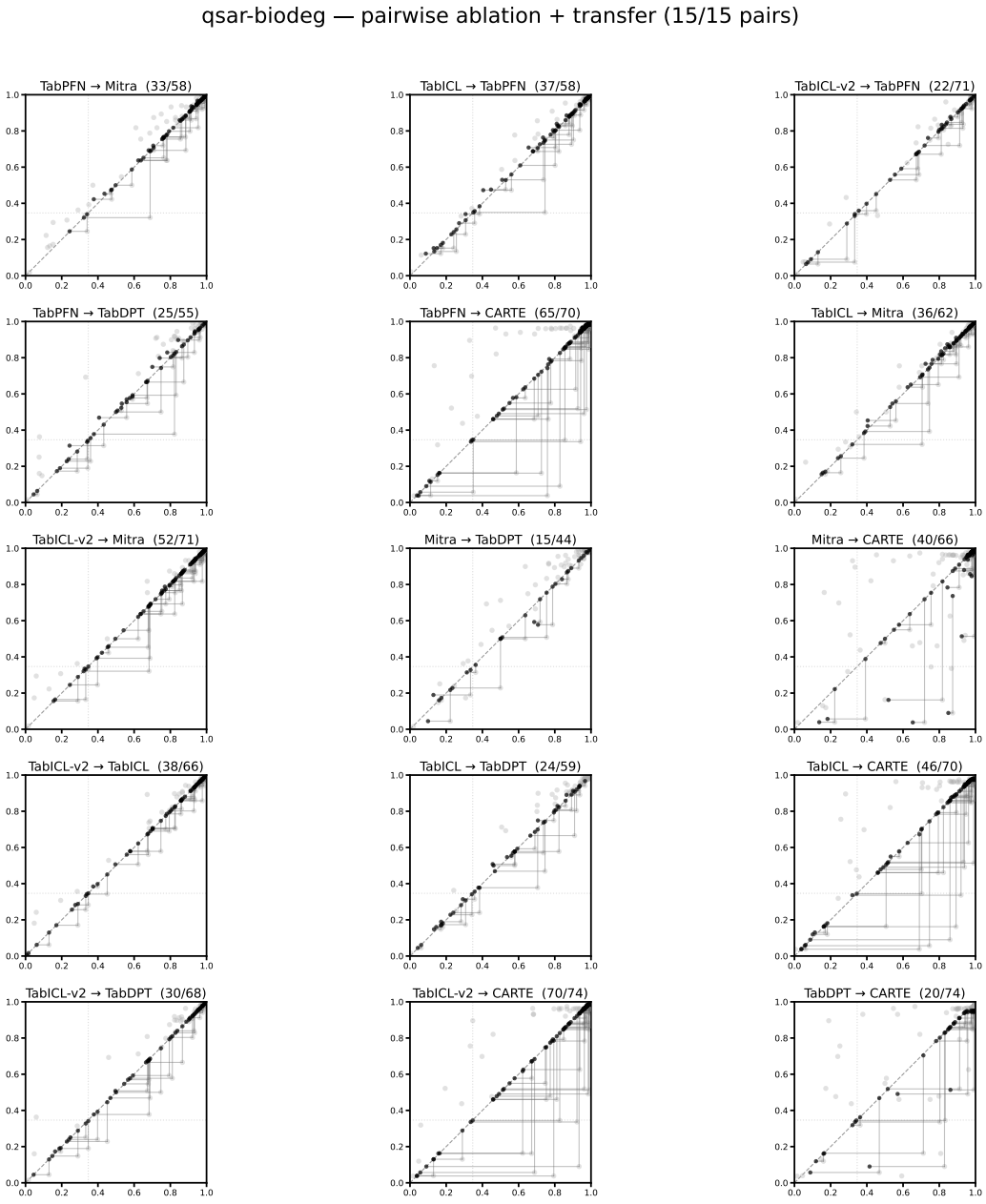


Figure 55: QSAR Biodegradability

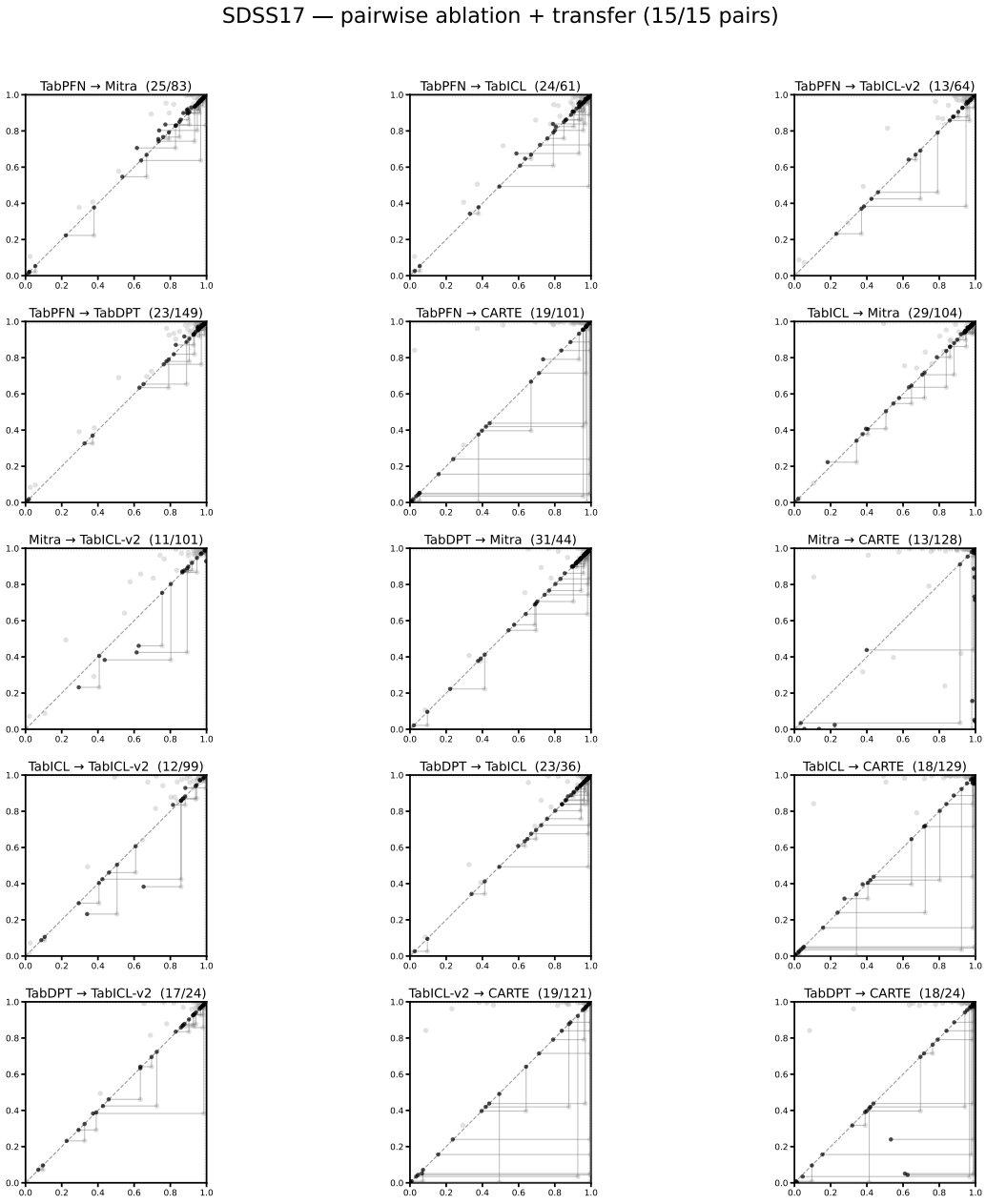


Figure 56: SDSS17

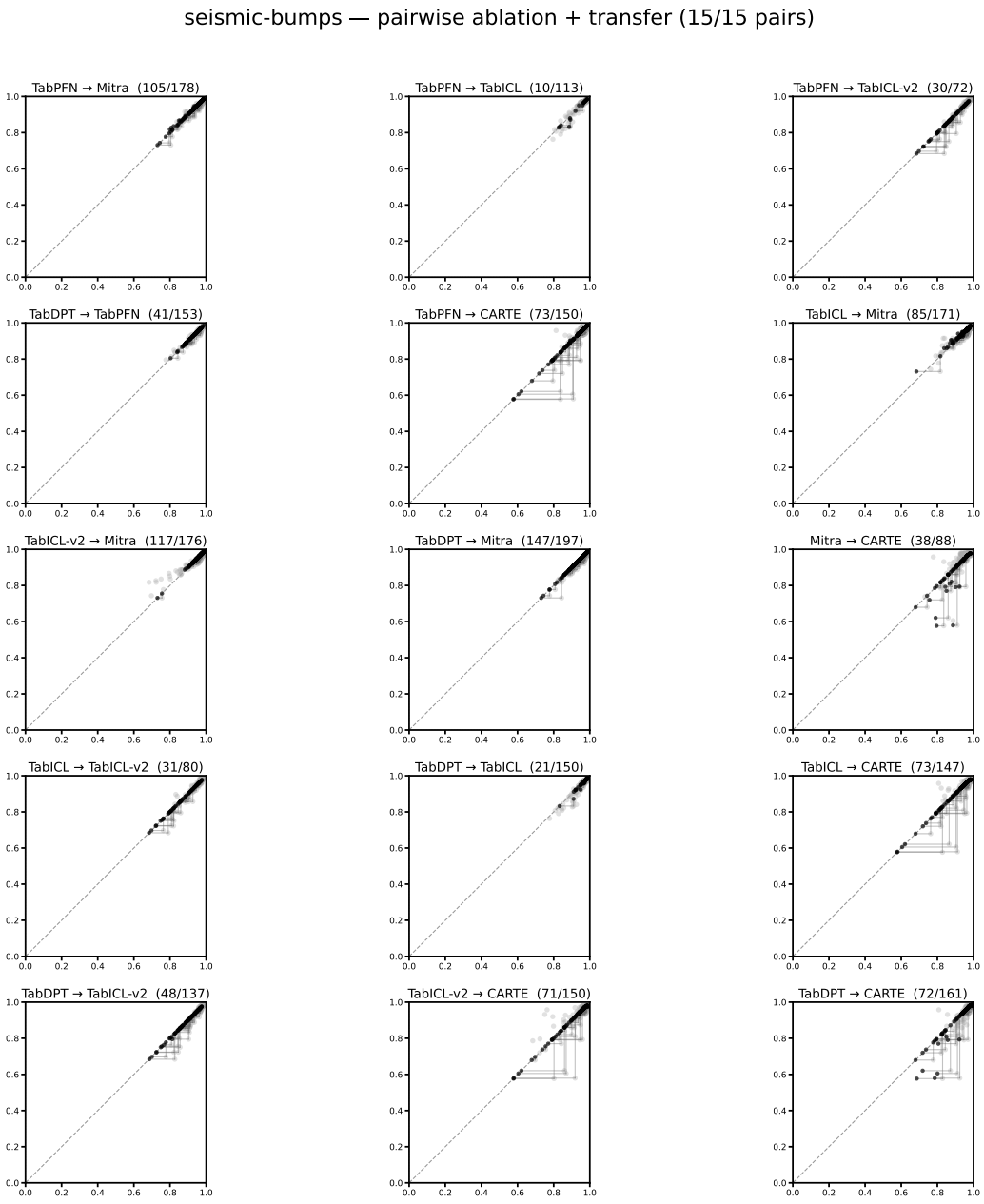


Figure 57: Seismic Bumps

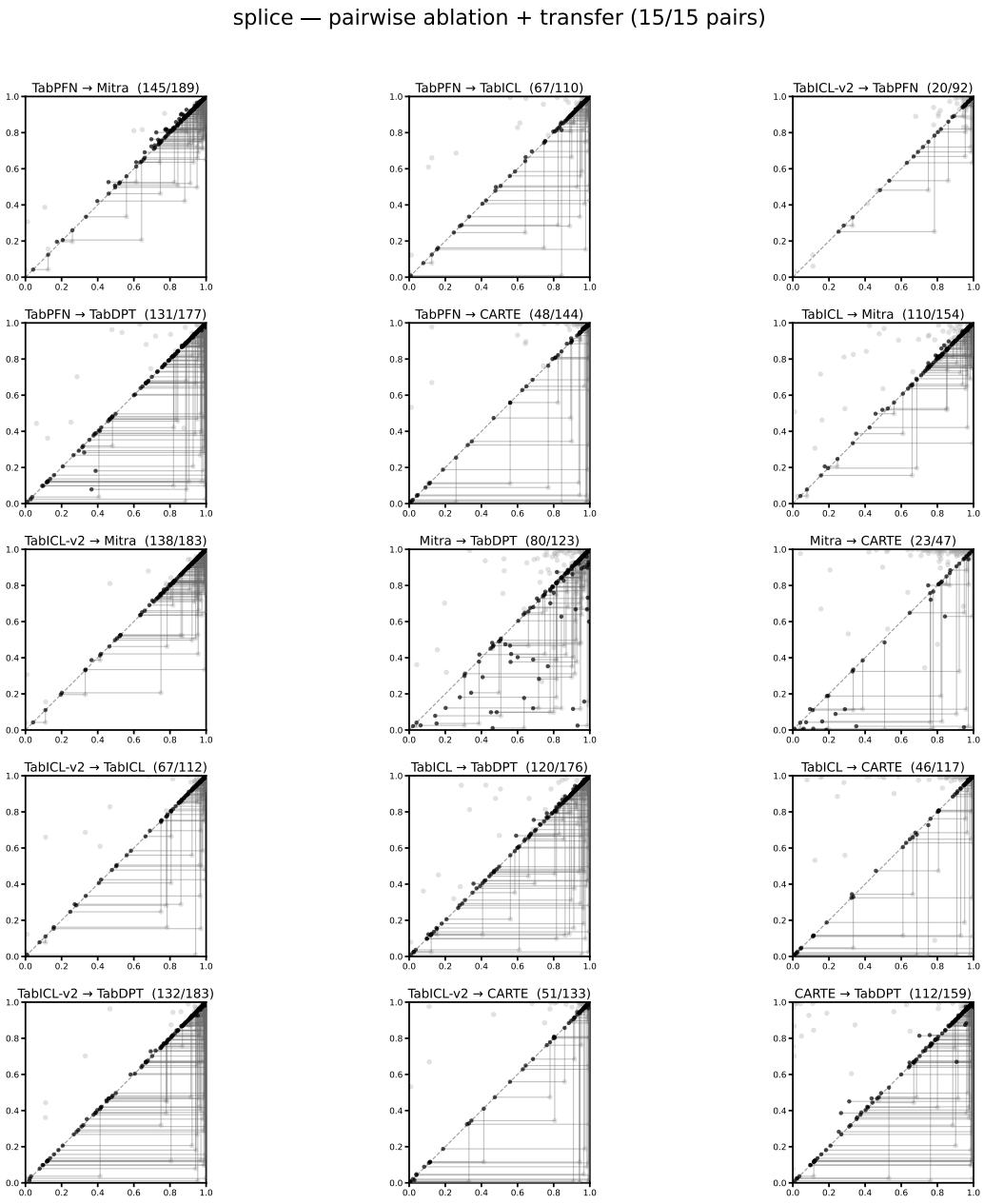


Figure 58: Splice

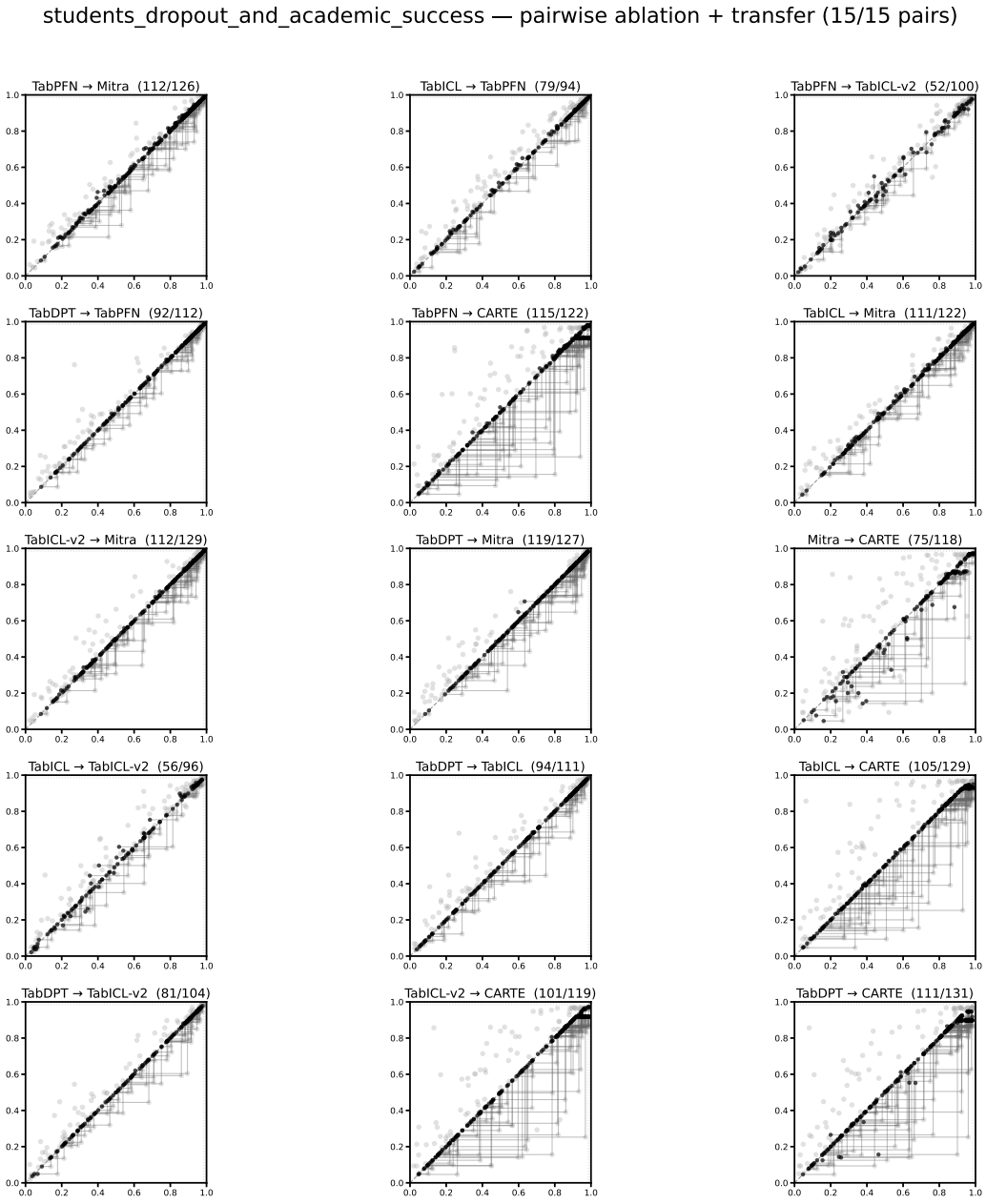


Figure 59: Students Dropout and Academic Success

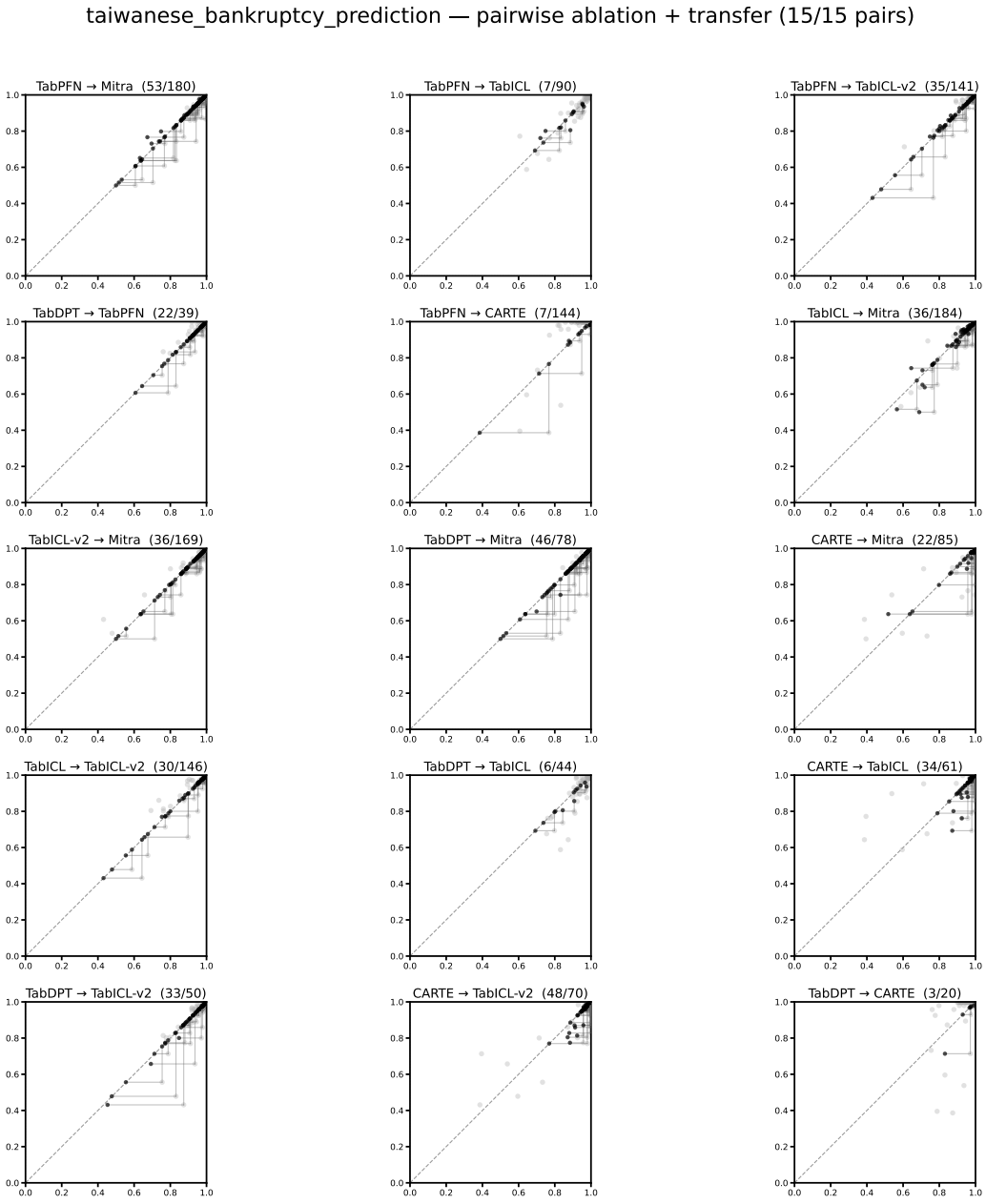


Figure 60: Taiwanese Bankruptcy Prediction

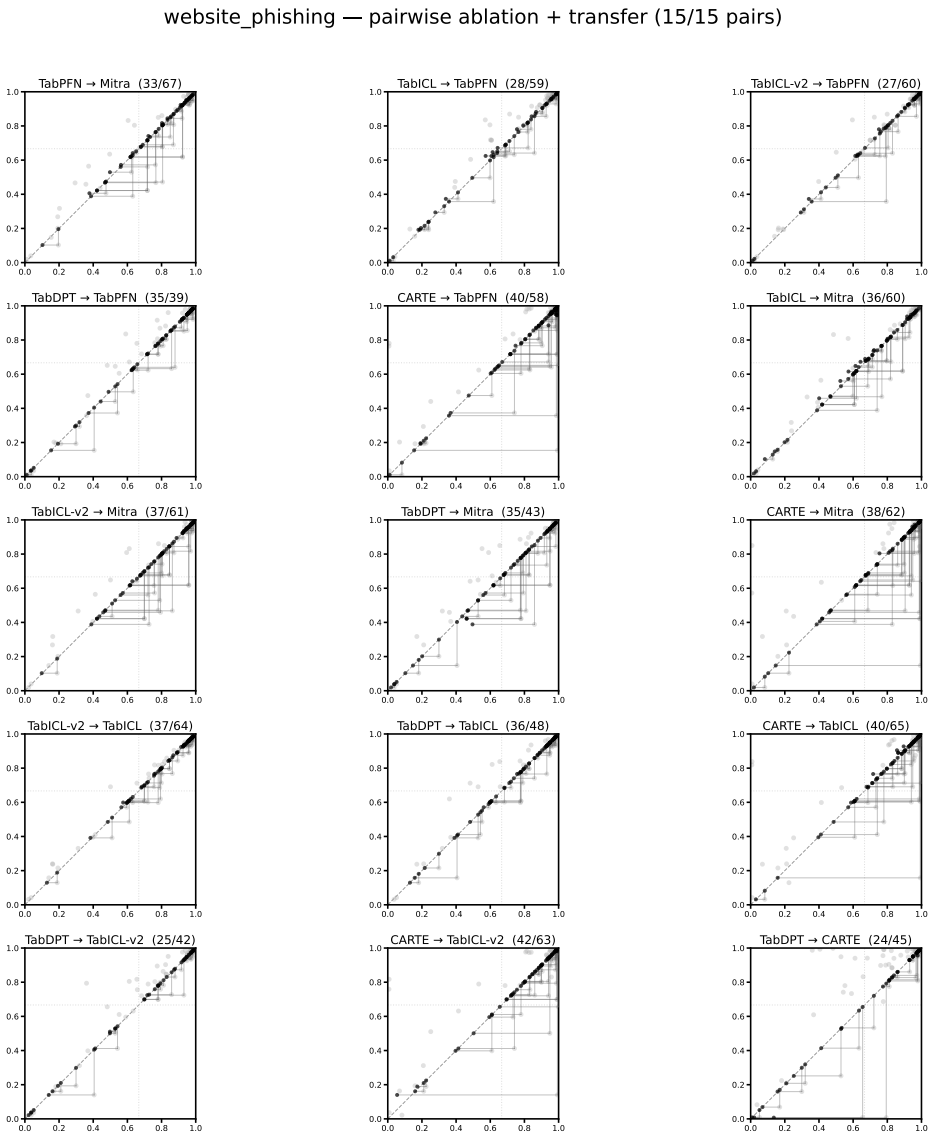


Figure 61: Website Phishing

I.2 REGRESSION

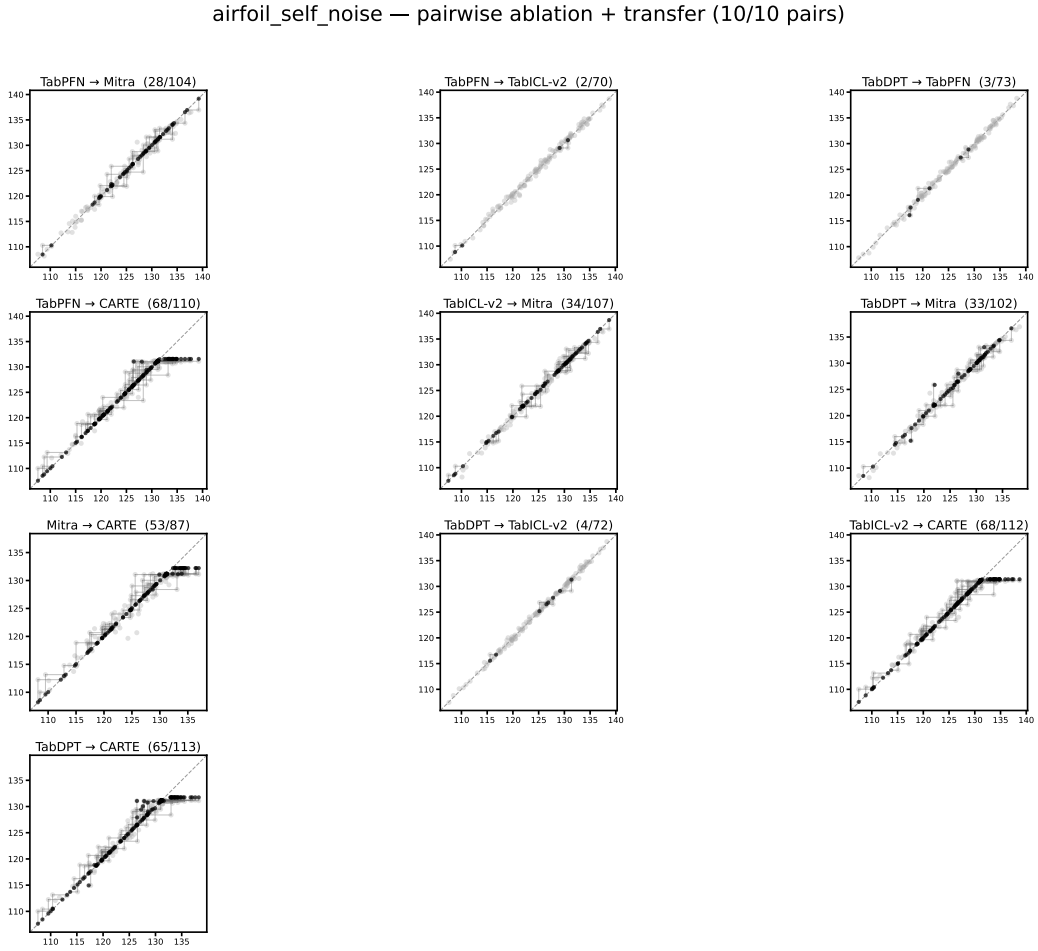


Figure 62: Airfoil Self Noise

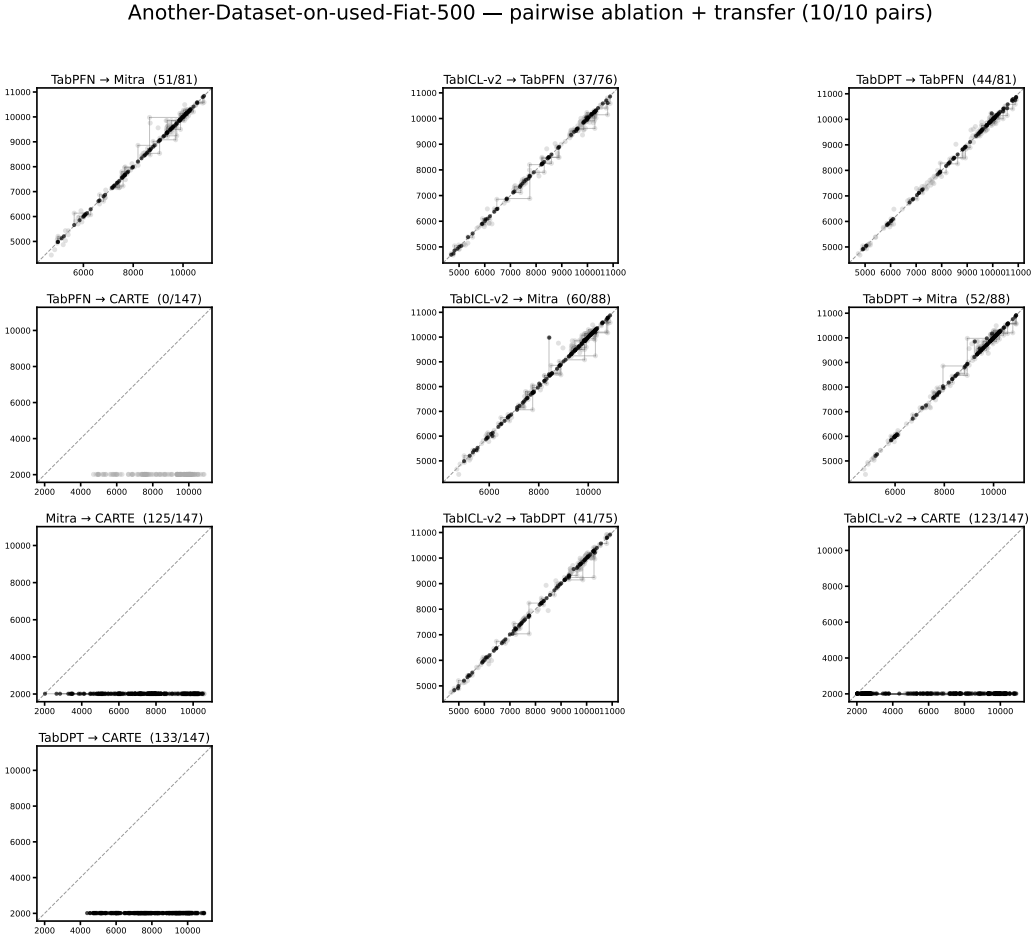


Figure 63: Another Dataset on Used Fiat 500

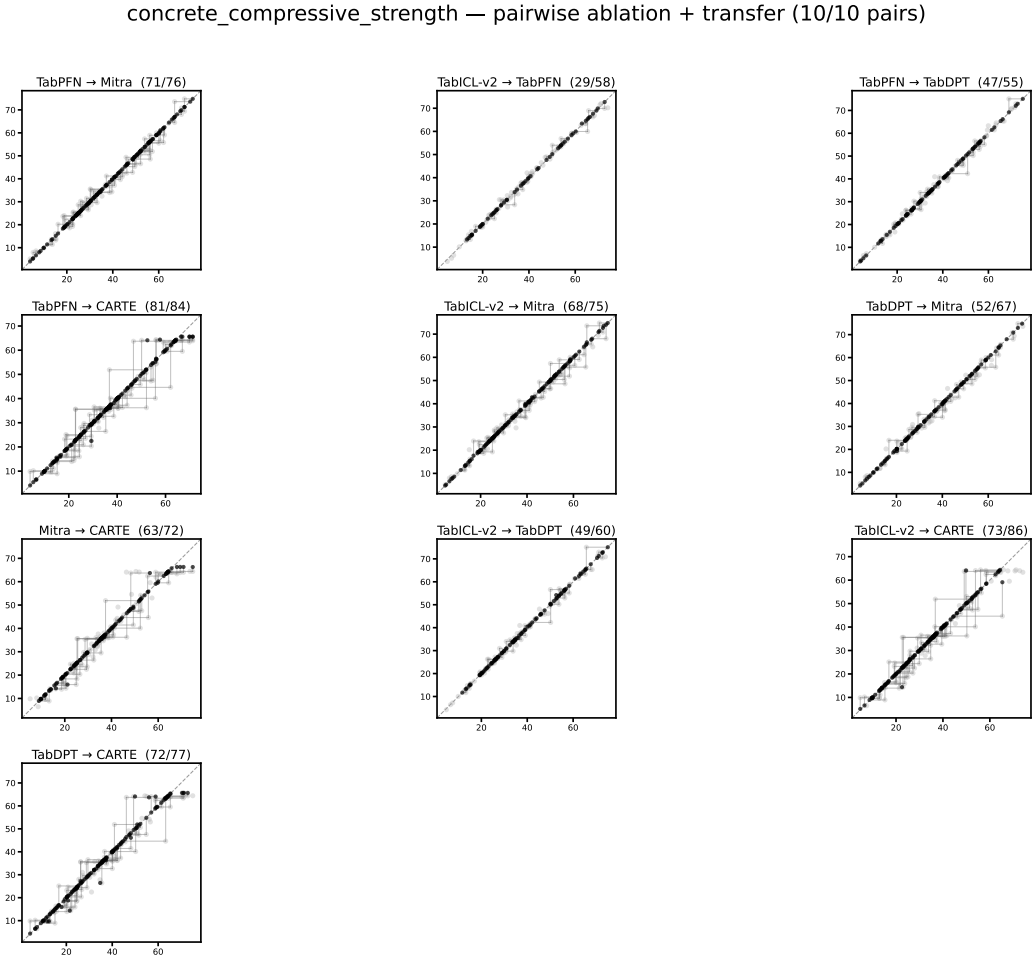


Figure 64: Concrete Compressive Strength

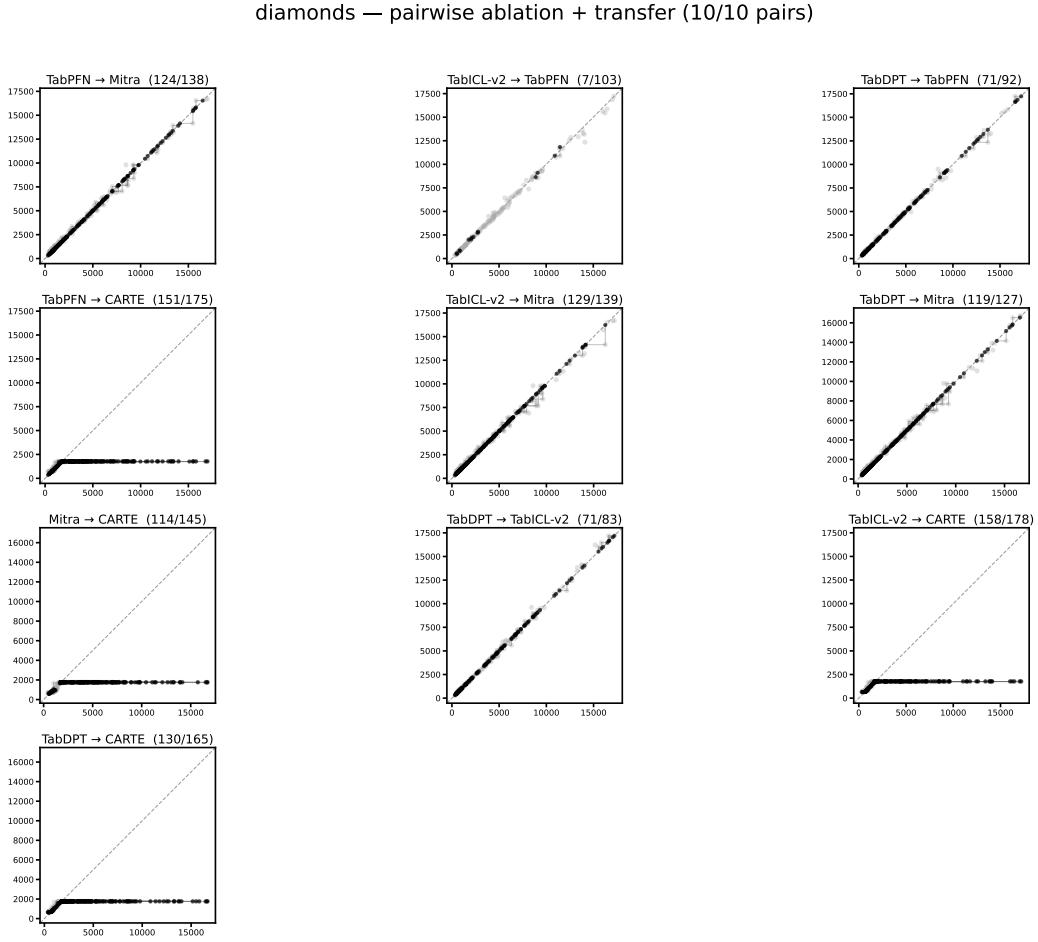


Figure 65: Diamonds

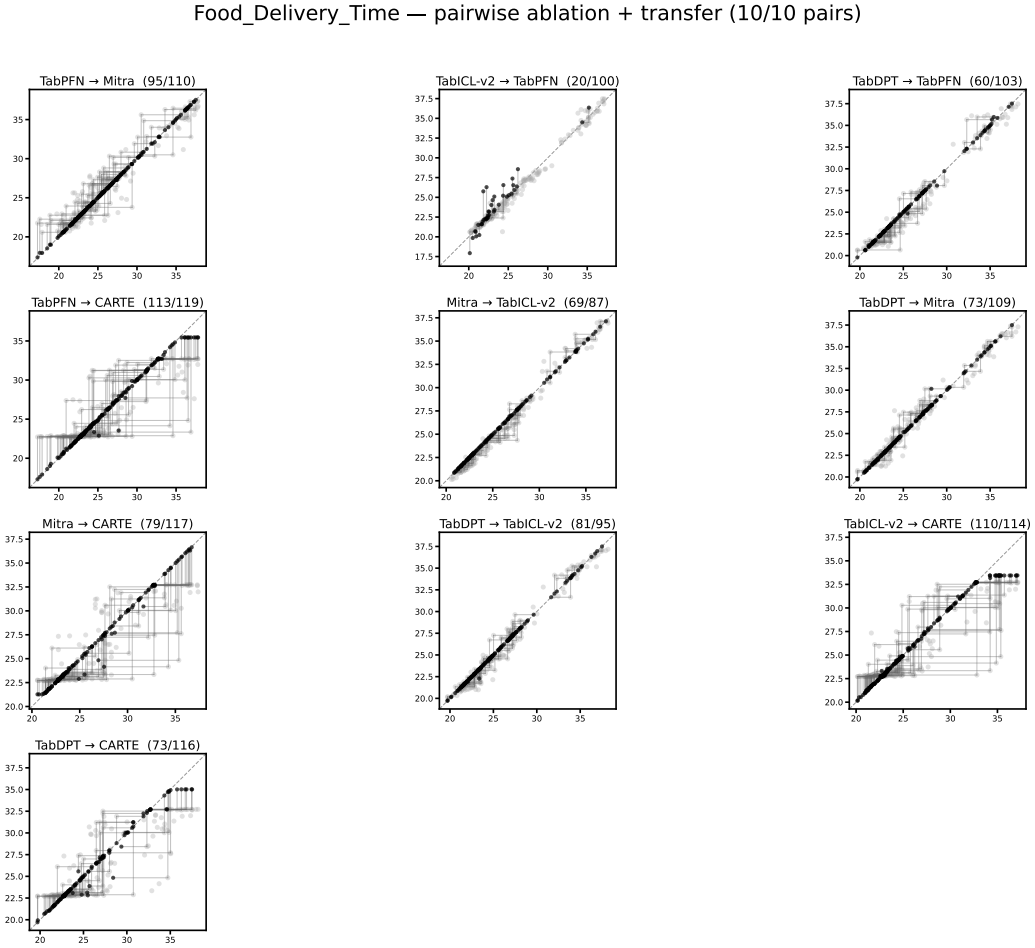


Figure 66: Food Delivery Time

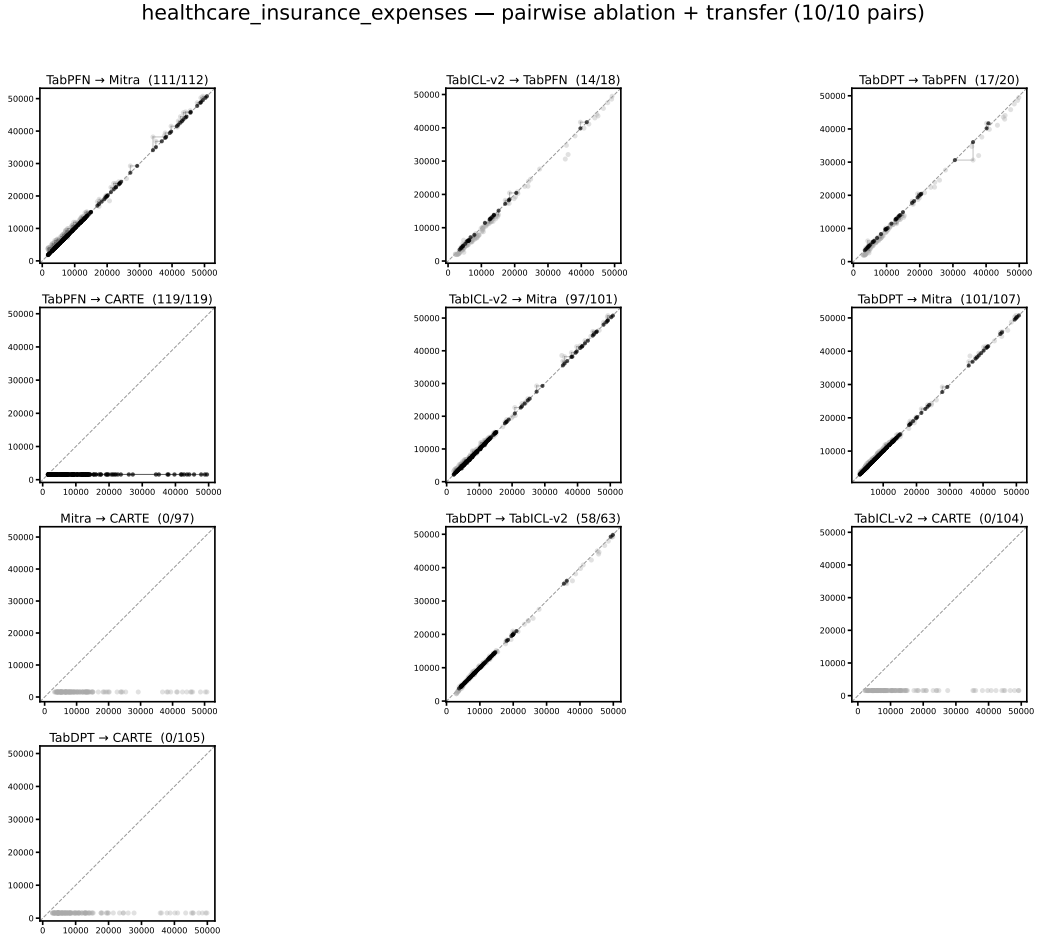


Figure 67: Healthcare Insurance Expenses

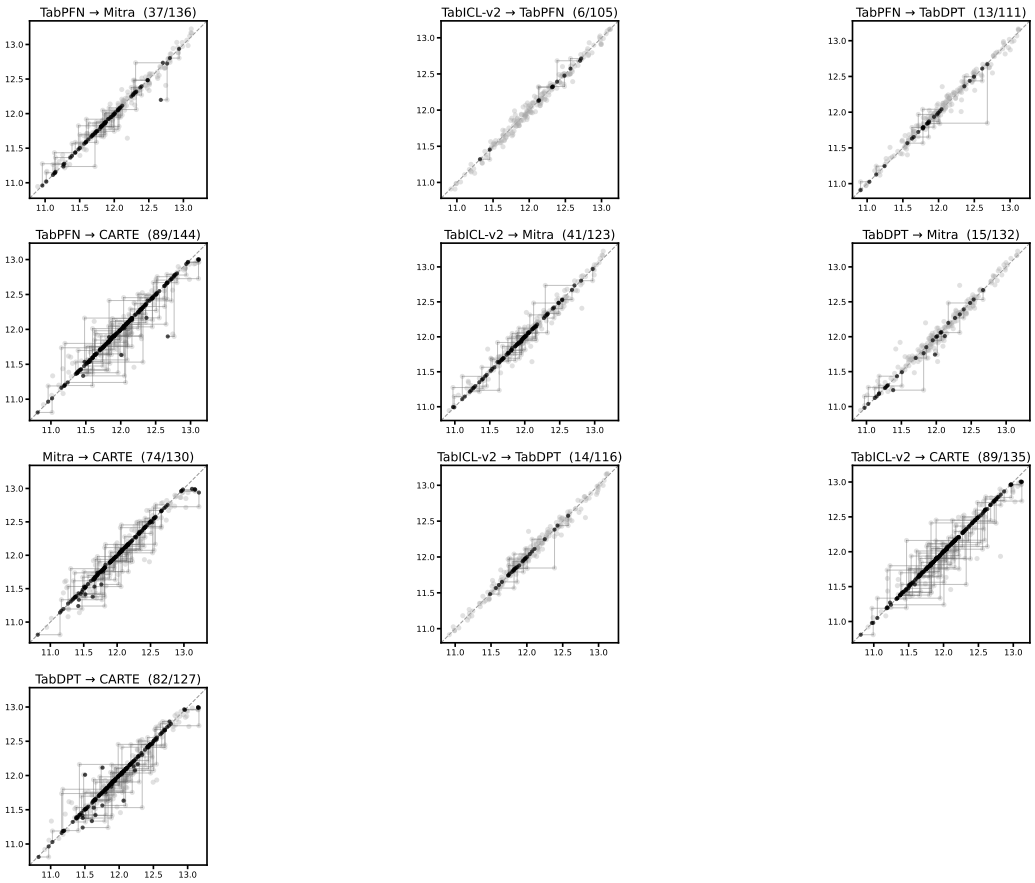


Figure 68: Houses

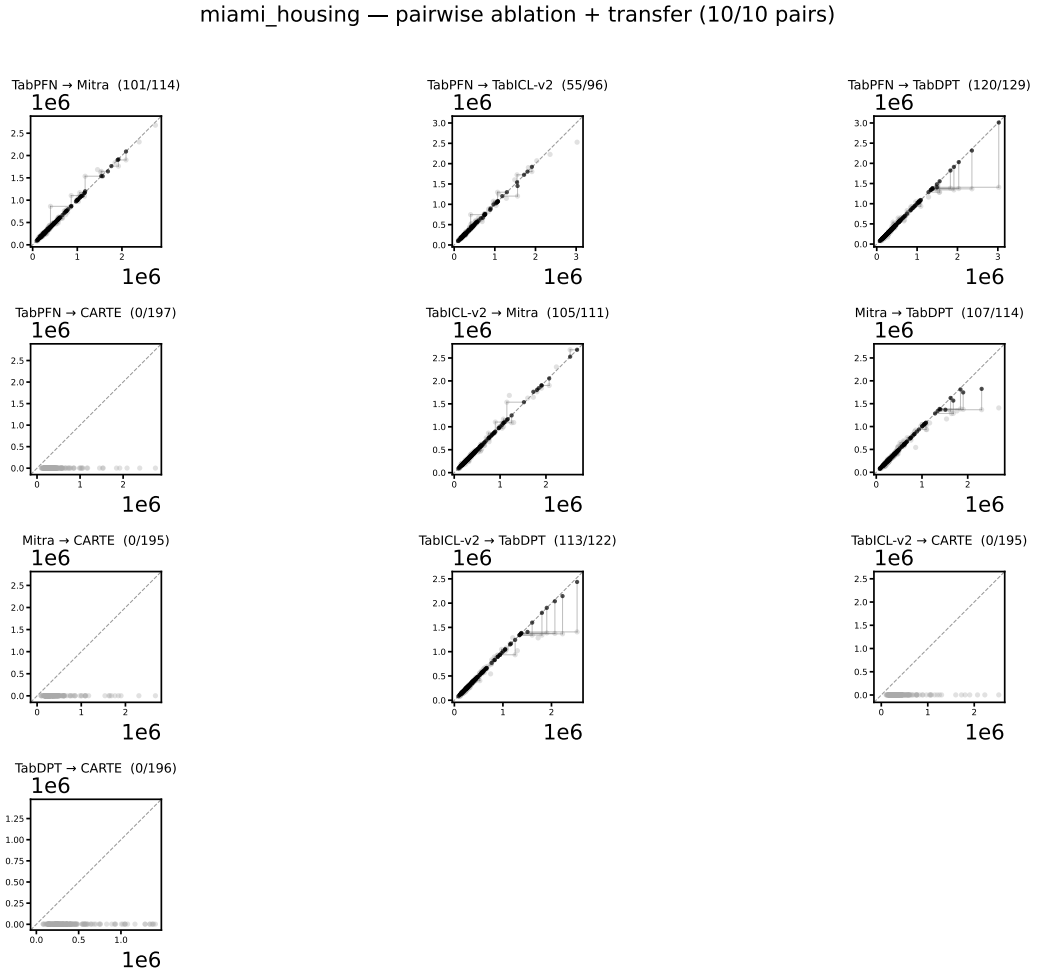


Figure 69: Miami Housing

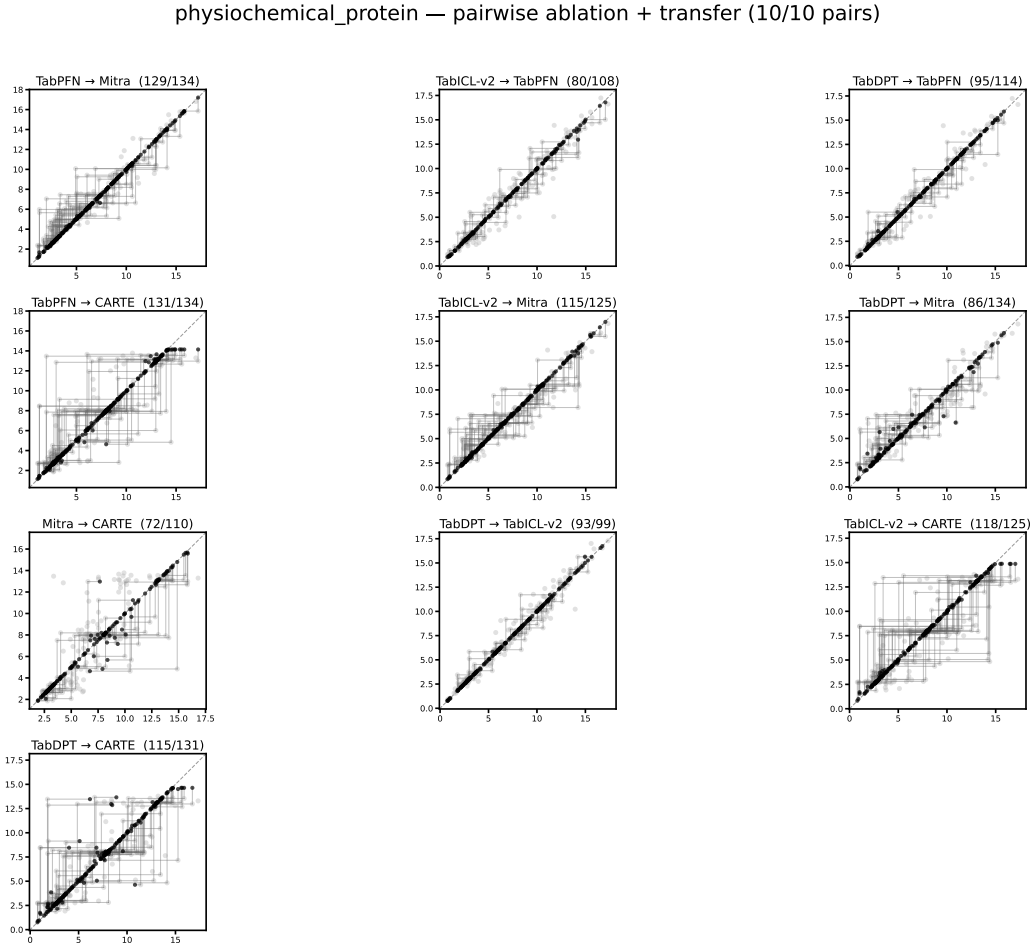


Figure 70: Physiochemical Protein

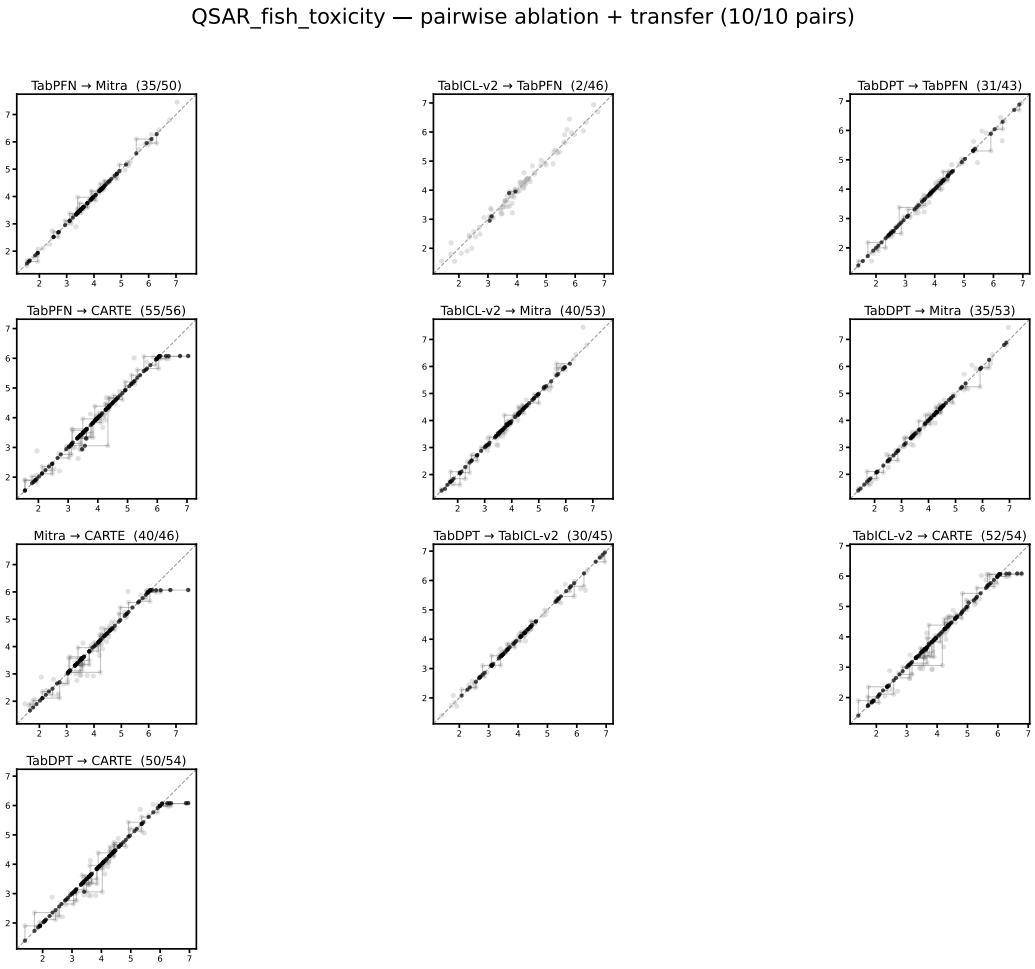


Figure 71: QSAR Fish Toxicity

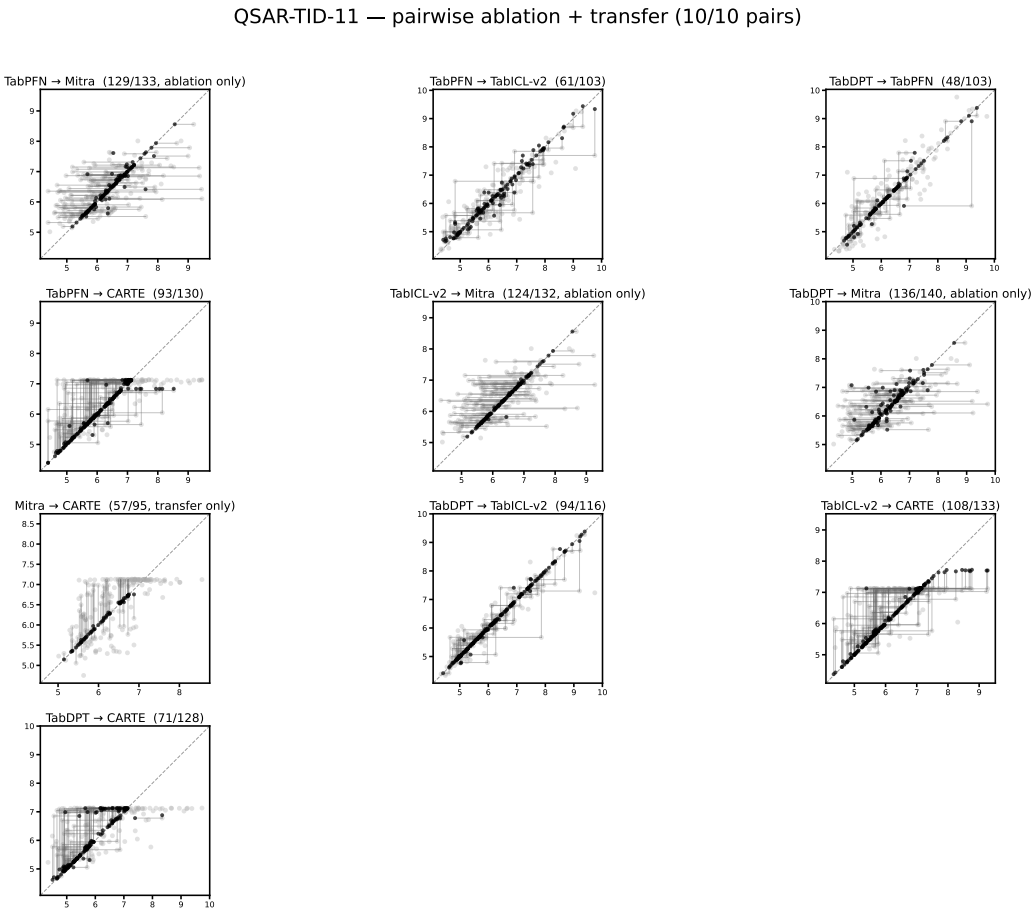


Figure 72: QSAR TID-11

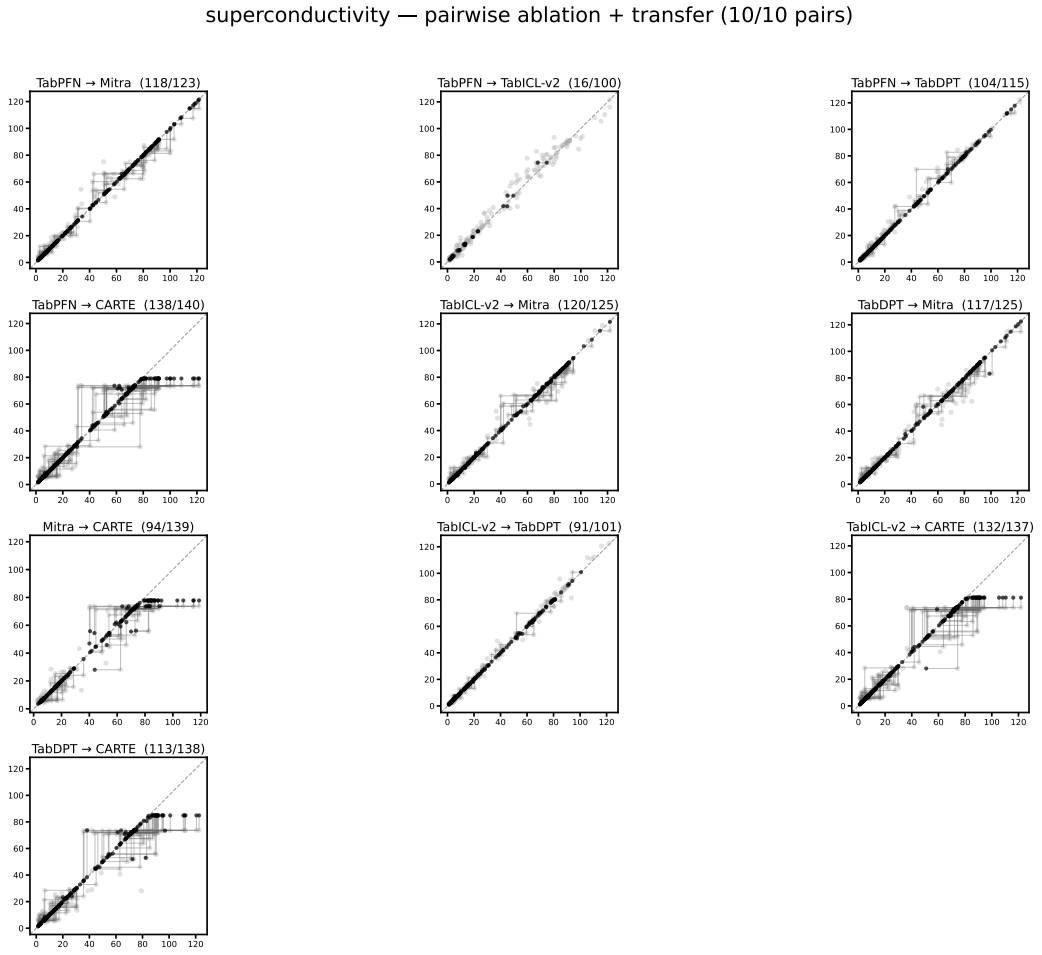


Figure 73: Superconductivity

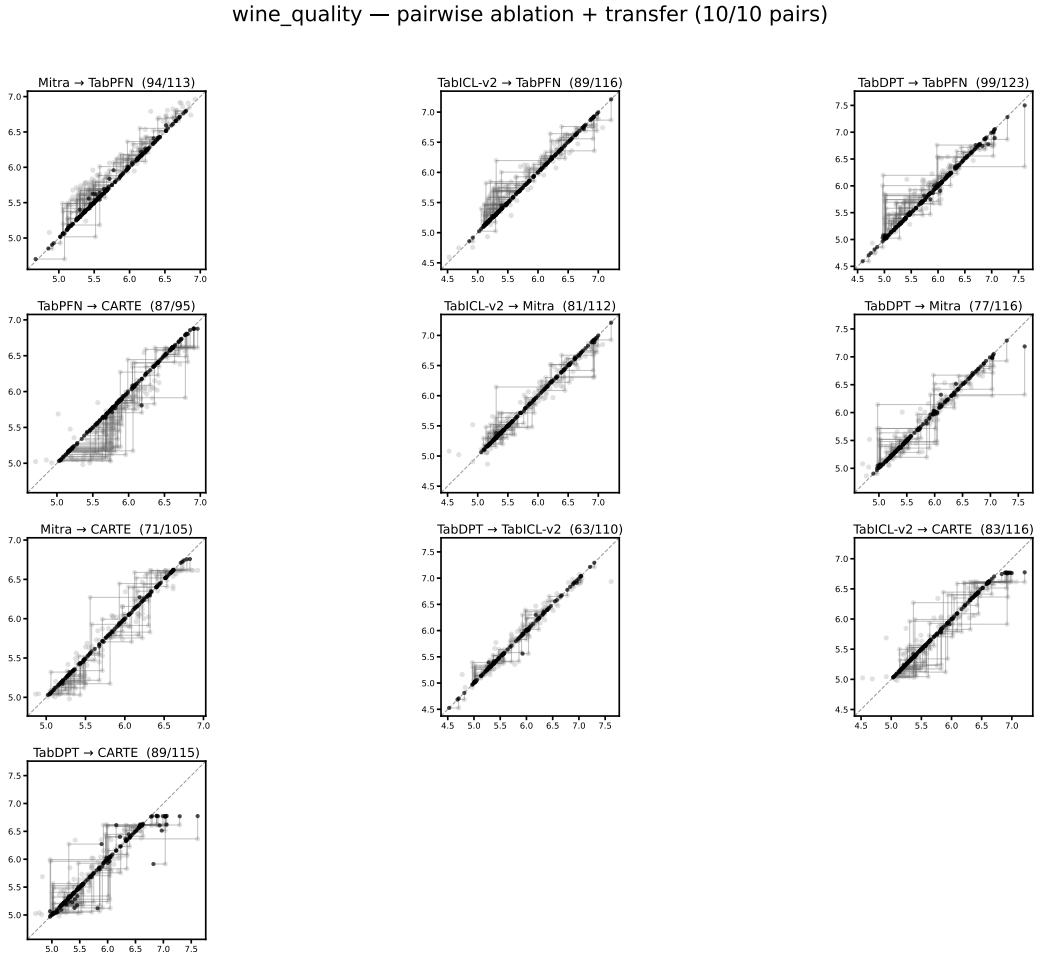


Figure 74: Wine Quality