CARVEPrep: Model-Aware Data Preparation for Tabular Foundation Model Post-Training

Anonymous authors
Paper under double-blind review

Abstract

Tabular foundation models such as TabPFN are post-trained on task tables, where each record both updates the parameters and serves as context for prediction. Preparing a dirty table thus amounts to allocating a data action to each record. Detecting an error does not show that fixing it helps the model, and action effects need not add, so these decisions should be judged by the trained model rather than by data correctness. We present CARVEPrep, which flags errors from dependency violations and label checks, trains a predictor on self-injected errors, without true values, to choose each record's action, and keeps a prepared table only when repeated short post-training shows that it beats the unchanged one. Feedback from the trained model then reprioritizes records for the next round. We prove that, under the same budget on changed records, removing repair or partial weights can strictly hurt. On six tables with injected errors, CARVEPrep averages 0.8119 accuracy, against 0.7343 for the unchanged data and 0.7853 for the strongest of fourteen baselines, which rewrites 70 times as many cells, and matches or beats every baseline on four of six tables. With data-specific evidence, the same actions carry over to sensor time series, where CARVEPrep ranks first on five of six fault scenarios, and to instruction tuning of Qwen2.5-1.5B, where held-out loss drops from 1.396 to 1.312. Tuning data and post-training hyperparameters together matches or beats tuning either alone on 11 of 12 tasks.

1 Introduction

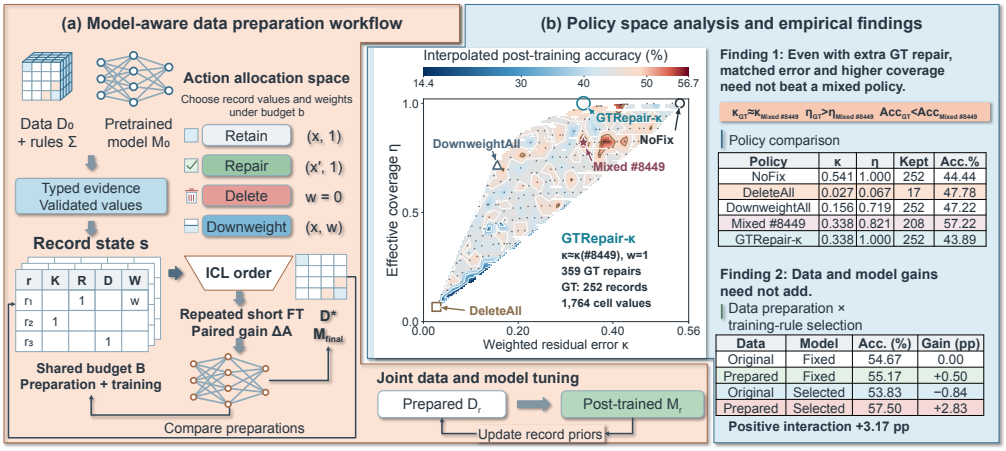
Data preparation determines what a model can learn from a task table. An error that is not repaired changes the associations the model learns, while deleting the record that holds it also discards useful observations (Zha et al., 2025). For tabular foundation models, deciding which records to repair, delete or retain affects the model in two ways. TabPFN predicts from a labeled support set supplied in context (Hollmann et al., 2025), and post-training fine-tunes its parameters on task data while keeping this in-context prediction (Rubachev et al., 2025). Each training record therefore enters both the parameter updates and the context used to predict new queries.

In practice, better data can yield gains that model tuning does not (Jarrahi et al., 2023; Jakubik et al., 2024). In a steel-defect detection case, cleaning the data alone raised accuracy from 76.2% to 93.1%, while model changes gave no gain (Ng, 2021). Yet automatic repairs can introduce errors, and higher data-level correctness need not yield better predictions (Ni et al., 2024; Abdelaal et al., 2023). Preparation and training are also coupled. Downweighting lowers a record's share of the loss and context but keeps its values, and the best training rule shifts once the data are prepared. The goal of preparation is therefore *the performance of the model trained on the prepared data*.

Allocating actions to records is a joint data-model decision. Repair changes values and can restore a correct one or write a wrong candidate. Deletion and downweighting act on the record weight. Deletion removes the record from training and context together with its valid information, while downweighting keeps it at reduced influence. Retention keeps both values and weight, errors included. The same action can help one record and hurt another, so improving a table means deciding *where to repair, where to reduce influence, and where to leave the data unchanged*.

Figure 1(b) illustrates why this action allocation must go through model feedback. **Finding 1.** Compared with the action-allocation policy, ground-truth repair, even though it uses 252 extra fully clean true values, reaches lower accuracy (43.89% vs. 57.22%). **Finding 2.** Selecting the training rule

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.



Policy	κ	η	Kept	Acc.%
NoFix	0.541	1.000	252	44.44
DeleteAll	0.027	0.067	17	47.78
DownweightAll	0.156	0.719	252	47.22
Mixed #8449	0.338	0.821	208	57.22
GTRepair-κ	0.338	1.000	252	43.89

Data	Model	Acc. (%)	Gain (pp)
Original	Fixed	54.67	0.00
Prepared	Fixed	55.17	+0.50
Original	Selected	53.83	−0.84
Prepared	Selected	57.50	+2.83

Figure 1: Why data preparation needs model feedback. (a) Model-aware data preparation. (b) The search surface reuses 9,681 distinct action-allocation policies on the real Flights dataset with TabPFN as the downstream model, computing their residual error rate and effective coverage rate with the corresponding post-training accuracy. Finding 2 crosses data preparation with fixed or reference-selected post-training configurations on the Beers dataset (Table 1).

alone lowers accuracy, preparing the data alone raises it, and selecting the rule on the prepared table raises it further. Therefore, the marginal gain of training-rule selection can change with data preparation, producing a positive interaction rather than a simple sum of independent gains. These two findings show that data-centric improvement is as important as model tuning, and that its key is how actions are allocated to records, whose effect can be judged only through model feedback (Appendix C.6).

Challenges. Model-aware allocation of data actions faces three difficulties.

(1) Evidence of error does not determine the best action. A dependency violation can flag an inconsistent value but cannot tell whether repair, deletion or downweighting will help the model. Prediction-based scores add task information but can miss errors that leave the joint distribution of labels and predictions unchanged, such as a value redrawn at random from its own column. The decision must link these signals to what each action does to the model.

(2) Interacting actions make the allocation space hard to search. Even with a fixed number of actions and a finite weight set, possible allocations grow exponentially with the record budget, and action gains need not add because records jointly determine training updates and prediction contexts. Ranking actions independently therefore cannot identify the best complete preparation.

(3) Downstream comparison is costly, noisy and model-specific. Each candidate requires model adaptation, so exhaustive post-training is infeasible, and a measured gain varies with training randomness and the finite evaluation sample. More candidates thus leave less budget to compare them reliably. The procedure must decide which candidates merit training and when the evidence justifies replacing the unchanged table, including when no intervention helps. It must also carry over to other downstream models and data types, whose evidence and metrics differ.

Contributions. We introduce CARVEPrep, which flags errors, decides each record's action, and lets the downstream model's feedback pick the complete preparation (workflow in Figure 1(a)).

(1) Modeling preparation as budgeted action allocation. We formulate data preparation as allocating retention, repair, deletion and downweighting actions under a record budget. This design needs only record values and weights, so it carries over to other models and data types. For a separable weighted least-squares family, we compute exactly how much an action set can gain under the budget and give an instance, stable under small perturbations, where removing repair or fractional weights strictly raises the minimum risk (Theorem E.1, Proposition E.5). We also prove that label- and prediction-based detectors can miss errors that dependency checks find (Proposition E.4).

(2) Record-wise action decisions learned without true values. Trained on errors injected into the working table, one regressor per action predicts each record's share of the resulting in-context

accuracy change, giving one allocation per cleaning fraction (Section 3.2). In-context screening narrows candidates before post-training. The regressors decide actions per record, but the accuracy of the whole preparation decides what is kept, because we prove that actions can interact, with a case where a deletion and a repair each raise the risk but together remove it (Proposition E.6).

(3) Joint data-model tuning driven by model feedback. Repeated short post-training compares candidates with the unchanged table, keeping one only if it passes the adoption rule. The trained model's held-out losses then reprioritize records for the next round (Section 3.3). Needing only measured scores, this feedback works for other models and metrics. With an independent acceptance sample, a held-out variant bounds the chance of adopting a preparation worse than the unchanged table by more than a margin, including after retraining (Proposition E.7, Corollary E.2).

(4) Evidence across data scales, data types and downstream models. On six tables with injected errors, CARVEPrep averages 0.8119 accuracy, against 0.7343 for unchanged data and 0.7853 for the strongest of fourteen baseline adaptations, which rewrites 70 times as many cells. CARVEPrep matches or exceeds every baseline on four of six tables, and on Adult pools up to 10,000 records it stays above NoFix by 1.7 to 9.2 points. With the same four actions and data-specific evidence, it ranks first on five of six sensor time-series fault scenarios (TEP), and with an LLM as the downstream model it lowers the held-out loss of Qwen2.5-1.5B on Alpaca instructions from 1.396 to 1.312. Tuning data and training rule together matches or beats tuning either alone on 11 of 12 tasks (Table 1).

2 Problem Definition

Given a dirty table D_0 with n records and d columns, dependencies Σ, and a downstream model M with a performance metric *Perf*, the goal is to allocate a data action to each record, under a record budget, so that M post-trained on the prepared table D_π performs as well as possible. Here *Perf* denotes population accuracy of the post-trained model. It is estimated on the reference set during search and on the clean test split for reporting, so every allocation is judged by model feedback rather than by data correctness.

Definition 2.1 (Suspect Set)**.** Let $S \subseteq \{1, \dots, n\}$ index the records eligible for intervention under the evidence or selection rule, and let $K = |S|$. Record r_i, $i \in S$, has evidence tuple $\langle t_i, s_i^{\text{FM}}, s_i^{\text{DB}}, \mathit{cand}_i \rangle$, where $t_i = \max_j t_{ij}$ is the maximum over its cell tiers t_{ij} (Definition B.5), s_i^{FM} is its record-wise prior score, $s_i^{\text{DB}} = (s_{ij}^{\text{DB}})_{j=1}^{d}$ its cell-wise structural evidence, and cand_i the validated candidate values for its cells.

Definition 2.2 (Four Actions)**.** The set $\mathcal{A} = \{\mathsf{retain}, \mathsf{repair}, \mathsf{delete}, \mathsf{downweight}\}$ contains the four data actions. Retain preserves values and sets $w_i = 1$. Repair changes specified cell values and sets $w_i = 1$. Delete sets $w_i = 0$. Downweight preserves values and sets $0 < w_i < 1$. Repair may be combined with downweight and is the only action that writes values, so it is the only source of repair error. These primitives represent every preparation that preserves record identities and columns with weights in $[0, 1]$, up to values on deleted records (Proposition E.1). CARVEPrep restricts repairs to validated candidate configurations.

Definition 2.3 (Allocation Policy)**.** For each $i \in S$, let $\mathcal{Z}_i$ contain distinct configurations comprising the unchanged record, indexed by 0, and the available validated repairs. A policy assigns a configuration $z_i \in \mathcal{Z}_i$ and a weight $w_i \in [0, 1]$ to each suspect record. All other records retain their values and weight one. At weight zero we use the canonical representation $z_i = 0$. The intervened set is $\mathcal{I}(\pi) = \{i \in S : z_i \neq 0 \text{ or } w_i \neq 1\}$. Combining repair with downweight counts as one intervened record.

For $f, \alpha \in [0, 1]$, set $b = \min\{K, \lceil fn \rceil\}$. The goal is

$$\sup_{\pi} \mathit{Perf}(M; D_\pi) \quad \text{s.t.} \quad |\mathcal{I}(\pi)| \leq b, \qquad \text{RepairError}(\pi) \leq \alpha, \qquad \sum_i w_i > 0, \tag{1}$$

where RepairError is the fraction of modified cells that differ from the truth, set to zero when no cell changes. The record budget b limits how many records an allocation changes, and α limits how often its repairs are wrong, since repair may additionally require costly acquisition or verification of true values. Proposition E.2 characterizes the resulting search space and shows that every feasible policy connects to the unchanged table by feasible single-record reversals. Proposition E.3 bounds its residual error and effective coverage. The budget applies to the record-wise allocations that CARVEPrep

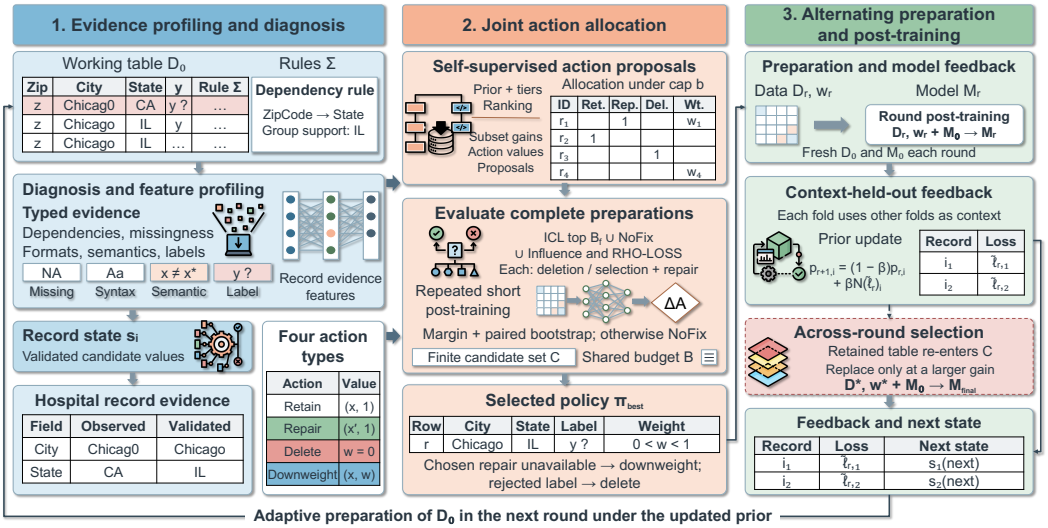


Figure 2: The CARVEPrep framework. (1) Typed evidence and validated values form record states. (2) Regressors trained on subset-average ICL gains decide actions under record budget b. Candidate preparations are screened by ICL, post-trained repeatedly on a short schedule and compared with NoFix by a gain margin and a paired bootstrap. (3) Each round restarts from D_0 and M_0, context-held-out losses update the priors, and the retained table is replaced only by a larger measured gain. The final model is retrained on the selected values and weights. The label Influence in Figures 2 and 4 denotes Ref. likelihood (Appendix C.8).

constructs. The final comparison also includes complete baseline preparations, which may exceed b, and judges every candidate by the same adoption rule (Section 3.2).

Definition 2.4 (Budgeted Alternating Preparation and Post-Training)**.** Preparation and post-training alternate so that feedback from each trained model reaches the next preparation. At iteration $r = 1, \ldots, R_{\text{loop}}$, we construct and compare preparations of the original table D_0 under prior p_r to obtain D_r, post-train from the pretrained M_0 to obtain an evaluation model M_r, and update the next prior from context-held-out losses. The iteration cost $cost_r$ includes preparation, candidate scoring, training and feedback. After selecting iteration $r^\star$, the procedure trains a final model $M^{\text{final}}(D_{r^\star})$ anew under a fixed final-training rule, and $cost_{\text{final}}(r^\star)$ covers this final training and evaluation. The objective is

$$\max_{R_{\text{loop}},\ 1 \le r^\star \le R_{\text{loop}}} \mathbb{E}_{\text{train}}[Perf(M^{\text{final}}(D_{r^\star}))] \quad \text{s.t.} \quad \sum_{r=1}^{R_{\text{loop}}} cost_r + cost_{\text{final}}(r^\star) \le B. \tag{2}$$

Here B is the total budget, and the expectation is over final-training randomness given the selected preparation. For a trained model instance, *Perf* is its population accuracy, and the final model is never scored on its training table. The output consists of the selected preparation and the newly trained model. Each allocation yields one prepared table, which is then trained on. With a zero allocation budget, training uses the unchanged D_0. The objective counts the cost of every stage, and the experiments use fixed iteration counts and short-training schedules (Appendix C.8).

3 The CARVEPrep Method

Evidence and learned estimates decide the record actions, and model feedback decides which complete preparation is kept. Figure 2 organizes CARVEPrep into three stages. Stage 1 turns the table into record states with typed evidence and validated candidate values (Section 3.1). Stage 2 decides an action for each eligible record under the record budget and adopts a complete preparation only when repeated short post-training measures a gain (Section 3.2). Stage 3 returns the held-out losses of the trained model to the next round (Section 3.3). Model feedback therefore enters twice, as measured gains that decide adoption and as losses that update record priorities.

Algorithm 1 Evidence-Prioritized Action Allocation and Two-Stage Adoption

Input: Working table with evidence, fraction grid F, reference set D_{ref}, learning setting
Output: Adopted values, actions and record weights
1: **Fit** repair, deletion and downweighting gain regressors from self-supervised perturbations
2: **Rank** records by $\prec_{\text{e}}$, rejected labels first, then evidence tiers plus the reference prior
3: **for** $f \in F$ **do**
4: Within the first $\lceil fn \rceil$ ranks, give each eligible record its action of largest positive estimated gain
5: **Retain** otherwise, **delete** rejected labels, and **downweight** a chosen repair without a validated value
6: Keep repaired values at reduced weight if evidence of error remains
7: **end for**
8: **Add** uniform, prior-guided, Ref. likelihood and RHO-LOSS candidates, and the unchanged table (NoFix)
9: **Rank** all candidates by ICL accuracy on D_{ref} and, **if post-training**, repeatedly short-train the top B_f, NoFix and all Ref. likelihood and RHO-LOSS candidates
10: **Adopt** the largest paired gain over NoFix passing the margin and bootstrap tests, else keep NoFix
11: **return** Adopted values, actions and weights

3.1 Evidence profiling and diagnosis

The action space separates changes to values from changes to influence. Stage 1 first fixes this action space and then collects the evidence that decides where to act. The operator class consists of maps $D \mapsto (D', w)$ that preserve rows and columns and assign weights $w \in [0, 1]^n$, with each output record determined by that record, its evidence and table statistics (Definition B.1). An output record either keeps or changes its values, and its weight is zero, strictly between zero and one, or one. These cases are exhaustive, so every output of the class is an allocation of retention, repair, deletion and downweighting and of their compositions (Definition 2.2, Proposition E.1). Uniform cleaning, reference-based selection and record-wise allocation are therefore comparable as complete preparations, while record creation and weights above one lie outside the class. A prepared table can be summarized by its residual error κ, the weighted count of erroneous cells relative to the table size, and its effective coverage η, the effective sample size of the weights divided by the record count (Definition B.2). Both require true values, so they describe the policy geometry of Figure 1(b) only after the fact, and allocation relies on the evidence below instead.

Evidence channels. With the action space fixed, evidence channels record likely wrong cells as evidence tiers (Definition B.5). Structural channels read the table itself. For a dependency $A \to B$, the records that agree on A form a group. When the group's majority B value carries enough support, the cells that disagree with it are marked and receive it as their repair candidate, while a group without a majority only marks the disagreement. Missing markers, domain constraints and typo checks mark further cells. A prediction-based channel checks labels instead, rejecting a label when its probability under a model fitted on the reference set does not exceed its marginal probability (Appendices D.1 and E.2).

The two kinds of channels carry different information. Proposition E.4 bounds the recall of any rule that reads only labels and model predictions by the selection fraction plus a total-variation term between the two conditional laws of its input. For a dependency under marginal resampling, these laws can coincide even for a Bayes-optimal model, while the intact majority of each group still identifies every changed cell, and the chance that the majority is wrong decays exponentially in group size. Neither kind of channel shows that acting on a record improves the post-trained model. Stage 2 therefore learns what each action is worth and keeps an action only under model feedback.

3.2 Joint action allocation

Record-wise estimates decide actions, and the accuracy of complete preparations decides adoption. Stage 2 turns record states into a preparation in three steps. It ranks records by their evidence, decides an action for each eligible record from learned action values, and adopts a complete preparation only when post-training measures a gain over the unchanged table (Algorithm 1). We call the allocation joint because an action is adopted only as part of a complete preparation whose effect is measured as a whole, so the same evidence can lead to a repair on one table and to no change on another. This split between record-wise decisions and whole-preparation adoption follows from how actions combine.

Why complete preparations are compared. For the separable weighted least-squares family of Appendix E.3, Theorem E.1 gives the minimum excess risk under at most b interventions as the unprepared risk minus the sum of the b largest per-record reductions, so the advantage of a larger action set is exactly computable. Under the same budget, repair adds attainable prediction endpoints and fractional weights fill the interval between deletion and those endpoints. The full set therefore separates strictly from sets lacking either, and the gap survives sufficiently small perturbations of the training features and the target means (Proposition E.5, Corollary E.1). The per-record reductions add there because each record determines its own prediction. Once records share a prediction, the reductions need not add. Proposition E.6 gives an instance in which deleting three mislabeled records and repairing one feature value each raise the excess risk, while the two together attain zero. A tabular foundation model predicts each query from the whole weighted context, so its training records share predictions. CARVEPrep therefore uses record-wise estimates to decide actions and lets the measured accuracy of complete preparations decide adoption.

Deciding record actions. The first two steps decide the actions. Records are ranked by the reference-conditional prior plus their evidence tiers, with rejected labels placed first. Within the first $\lceil fn \rceil$ positions, records carrying suspect cells or rejected labels are eligible and the rest are retained, which enforces the record budget of Equation (1) by construction (Appendix D.2). Records with rejected labels are deleted. Every other eligible record receives the action with the largest positive estimated gain, is retained when no estimate is positive, and is downweighted when its selected repair has no validated value. The action values are learned on the working table without true values. Each perturbation trial injects extra errors into part of the table, applies one action to a random eligible subset T and measures the change $\Delta_a(T)$ in in-context accuracy on held-aside records. Each treated record takes the subset average $\Delta_a(T)/|T|$ as its target, and one regressor per action learns to predict this target from record features (Appendix D).

Adoption by model feedback. The last step adopts a complete preparation. The candidates are the allocations for every cleaning fraction, the prior-guided and uniform strategies, candidates built from two reference-based selection scores, and the unchanged table. In-context accuracy on the reference set screens them, and the leading candidates are post-trained repeatedly on a short schedule as specified in Algorithm 1. Their per-record correctness is compared with that of the unchanged table on the same reference records by a paired bootstrap, which also reads the dispersion of record-level differences, so mean gain alone does not decide adoption. The candidate with the largest mean gain is adopted when it passes the gain margin and the win-probability level, and the unchanged table is kept otherwise. A cell change is therefore accepted only when the preparation containing it is measured to be better (Appendix D.3).

3.3 Alternating preparation and post-training

Model feedback revises which records merit intervention. After adoption, Stage 3 returns the losses of the trained model to the next preparation. The adopted values and weights determine the training loss and the context sampling, and the loss of each record when it is held out of the context updates its prior for the next iteration (Definition 2.4). Records ranked highest by the updated prior are treated as having rejected labels, so a later iteration can mark label errors that no cell channel marked. Each iteration prepares D_0 again under the current prior and trains from M_0, so every record, including one deleted earlier, is reassigned. The retained preparation re-enters the comparison and is replaced only by a larger measured gain, and the final model is trained on the preparation retained across iterations (Appendix D.4, Algorithm 3).

The adoption rule so far reuses the development reference set for both search and comparison. Because the reference set has already been used to select preparations, gains on that set must be independently tested on data excluded from the search to obtain the statistical guarantee below. When an independent i.i.d. acceptance sample is available, a held-out variant freezes the candidate preparations and evaluation models on development data and applies a betting test to that sample (Waudby-Smith & Ramdas, 2024). Proposition E.7 bounds by δ_2 the probability of adopting a frozen model whose population score falls more than ϵ below that of the frozen reference. Corollary E.2 transfers this bound to the retrained model when a separate independent unlabeled sample bounds the prediction disagreement between the frozen and retrained models. When independently verified records are available, they can also calibrate the repair risk of future changes (Lemma E.1, Appendix D.5).

4 Experimental Evaluation

4.1 Setup

(1)**Data and downstream models.** The experiments cover three data types and two downstream models. We use six tabular datasets, six sensor time-series fault scenarios of the Tennessee Eastman process (TEP) and Alpaca instruction data (Table 8, Appendix C.8). Missing values, typos, dependency violations and label noise are injected into clean versions at mixture rate $\rho = 0.4$, which gives cell-level truth for repair errors. The downstream model is TabPFN-v2 for tables and time series, post-trained with record weights in both loss and context sampling and also evaluated in context, and Qwen2.5-1.5B-Instruct with LoRA for text (Table 7). (2)**Metrics.** We report the test accuracy of the final model, the held-out response loss on Alpaca, the repair error rate, and the numbers of affected records and rewritten cells (Appendix C.8, Table 9). (3)**Baselines and settings.** Table 1 compares CARVEPrep with fourteen baseline adaptations, five uniform strategies and clean training as an oracle, covering cleaning, detection, selection and foundation-model preparation (Table 2, Appendix C.8). All methods share the data splits and the downstream training rule, and methods that use reference information receive the same clean reference set. It supports prior fitting, candidate comparison and early stopping but never enters final gradient updates. (4)**Studies.** Beyond the main comparison and its per-task gains (Figure 4), error-rate sweeps from 0 to 0.5 test both post-training and in-context prediction (Figure 3, Appendix C.1). Ablations contrast the full method with twelve pipeline variants (Figure 5, Appendix C.2) and with its PriorOnly and StructuralOnly versions (Figure 3), while nested candidate sets and an evidence-recall comparison isolate the roles of the candidate set and of structural evidence (Table 3, Appendix C.1). Appendix C.3 reports variation across training runs, the selected actions and their group statistics, training-rule selection before and after preparation, the feedback loop and held-out acceptance (Figure 6, Tables 4 and 5). Scale studies enlarge the Adult and Tax pools to 10,000 records and shrink the reference set (Table 6, Appendix C.4). Appendices C.6 and C.7 give the motivating experiments and the quality-state screening.

4.2 Results and Findings

Where changes land matters more than how many are made. CARVEPrep exceeds Jellyfish by 2.66 points and recovers 55% of the gap between NoFix and clean training (Table 1). Across all three training runs on the same split and corruption, mean accuracy is 0.8028 for CARVEPrep, 0.7840 for the strongest baseline and 0.7333 for NoFix (Appendix C.3). Jellyfish rewrites 70 times as many cell values as CARVEPrep, with a repair error rate 6.8 times as high (Appendix Table 9). Yet Flights, where 67% of the cells it modifies differ from the true values, is the one table on which it beats CARVEPrep. Data-level correctness and model accuracy therefore come apart, as they do for the ground-truth repair in Figure 1(b), which is why CARVEPrep chooses among the deletions of selection methods such as RHO-LOSS, repairs and unchanged data by model feedback. No uniform action transfers across tables either. DeleteAll is the best uniform strategy in the Flights motivating experiment (Appendix C.6) but has the lowest six-table mean among them (Table 1).

The same preparation helps both routes by which records reach the model. Without injected errors, NoFix is selected on all six tables, so the procedure leaves clean data untouched. From $\rho = 0.4$ to $\rho = 0.5$, the mean gain over NoFix more than doubles. The same preparation improves both parameter adaptation and in-context prediction on every table, with larger gains when the prepared table serves as in-context support (Figure 3(a–e)).

Which action helps depends on the error type. Every adopted preparation in the main comparison deletes records, and Adult's also repairs verified feature values. With label errors present, the downstream comparison favors removing label-rejected records over feature repairs that would leave the labels wrong. With feature errors alone at the same rate, NoFix is close to clean training and three of six tables remain unchanged (Figure 3(f)). With intact labels, the test-best full-action candidate exceeds the test-best retain-and-delete candidate on half the tables, using fractional weights on Flights and verified repair with deletion on Tax (Appendix Figure 6(b)). These retrospective comparisons expose the capacity described by Proposition E.5. Whether this capacity is realized depends on the adoption step.

Record-wise allocation and structural evidence drive the gain. Against a restricted candidate set of NoFix and two selection candidates, the complete candidate set gains 4.28 points under the

Table 1: Accuracy at $\rho = 0.4$ on six tabular datasets and six TEP fault scenarios. CARVEPrep-Cert reserves 40% of the reference set for held-out acceptance and includes the Jellyfish preparation as a candidate. Circled method numbers rank the 21 non-oracle methods by tabular mean. Circled cell numbers mark their top-five column ranks. Bold shaded cells mark column-best values among non-oracle methods and separately within the final block. The last block compares original and prepared tables under fixed and individually selected training rules. These runs include training-rule search and do not enter the method ranking. Its first two rows give NoFix and CARVEPrep evaluated within this search. Tabular method-comparison cells report reference-selected results from three training runs. Appendix C.3 specifies tie-breaking and reports three-run means.

		Tabular datasets							TEP scenarios						
	Method	Adult	Beers	Breast	Flights	Hosp.	Tax	Mean	F01	F04	F05	F07	F12	F14	Mean
Uniform strategies															
⑩	NoFix	0.74	0.55	④0.95	0.45	0.86	④0.86	0.73	0.87	0.89	④0.90	④0.90	0.91	③0.90	0.90
⑤	RepairAll	0.74	0.55	④0.95	0.54	0.88	④0.86	0.75	0.89	0.90	④0.90	④0.90	0.89	③0.90	0.90
⑳	DeleteAll	0.74	④0.55	④0.95	0.43	0.31	0.50	0.58	0.90	0.90	④0.90	④0.90	0.90	③0.90	0.90
⑰	DownweightAll	0.74	0.54	③0.96	0.44	0.51	④0.86	0.67	⑤0.94	0.90	④0.90	④0.90	0.90	③0.90	0.91
⑲	RandomDelete	0.75	0.54	0.83	0.51	0.28	②0.87	0.63	0.83	0.76	④0.90	④0.90	⑤0.91	③0.90	0.87
Cleaning															
⑥	ActiveClean	0.74	②0.55	②0.97	0.44	②0.93	④0.86	0.75	0.88	0.89	④0.90	④0.90	0.90	③0.90	0.89
⑨	Picket	0.74	0.55	0.94	0.45	④0.90	④0.86	0.74	0.59	0.90	④0.90	④0.90	0.90	②0.93	0.85
⑭	Cleanlab	0.71	0.54	0.90	0.56	0.84	0.73	0.71	0.80	0.60	0.50	0.87	0.84	⑤0.56	0.70
⑪	PriorAlign	0.74	0.54	④0.95	0.45	0.84	④0.86	0.73	0.87	0.89	④0.90	④0.90	⑤0.91	③0.90	0.90
④	Raha+Baran	0.74	②0.55	0.94	0.55	③0.91	③0.86	0.76	0.83	⑤0.94	④0.90	④0.90	0.90	③0.90	0.90
⑧	BoostClean	0.74	⑤0.55	③0.96	0.45	⑤0.89	①**0.87**	0.74	④0.95	0.90	④0.90	④0.90	0.89	③0.90	0.91
③	Jellyfish	0.74	0.48	③0.96	②0.77	③0.91	⑤0.84	0.79	0.83	④0.96	④0.90	④0.90	0.90	③0.90	0.90
Selection															
⑫	DSIR	0.74	0.55	④0.95	0.45	0.77	④0.86	0.72	0.90	0.90	④0.90	④0.90	0.90	③0.90	0.90
㉑	EL2N	0.37	0.23	0.50	0.41	0.31	0.46	0.38	0.90	0.90	④0.90	0.53	0.83	0.19	0.71
⑮	RHO-LOSS	④0.82	0.54	①**0.97**	④0.59	0.43	0.80	0.69	②0.96	①**1.00**	③0.97	②1.00	③0.94	③0.90	0.96
⑦	Ref. likelihood	③0.83	①**0.56**	②0.97	0.57	0.67	②0.87	0.74	③0.95	②0.99	②0.97	③0.99	④0.92	③0.90	0.96
⑯	Data-OOB	⑤0.76	0.54	0.89	0.49	0.68	0.79	0.69	0.58	0.69	0.68	0.68	0.70	0.51	0.64
⑱	Herding	0.74	0.55	0.94	0.49	0.28	②0.87	0.64	0.74	③0.98	④0.90	0.89	0.76	③0.90	0.86
⑬	GraNd	0.69	0.54	0.94	⑤0.58	0.74	0.83	0.72	0.67	0.69	⑤0.71	⑤0.90	0.78	④0.57	0.72
Ours															
①	CARVEPrep	②0.84	③0.55	①**0.97**	③0.67	①**0.97**	①**0.87**	**0.81**	③0.95	①**1.00**	①**0.98**	①**1.00**	①**1.00**	①**1.00**	**0.99**
②	CARVEPrep-Cert	①**0.84**	0.55	⑤0.95	①**0.77**	0.84	④0.86	0.80	①**0.98**	①**1.00**	③0.97	①**1.00**	②0.96	①**1.00**	0.98
Oracle															
	Clean (oracle)	0.85	0.57	0.97	1.00	0.99	0.87	0.88	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Two sides of the decision															
	Untouched	0.74	0.55	0.97	0.45	0.86	0.86	0.74	0.94	0.88	0.90	0.90	0.91	0.90	0.91
	Data only	0.84	0.55	**0.97**	0.67	**0.97**	**0.87**	0.81	0.95	**1.00**	**0.98**	**1.00**	**1.00**	**1.00**	0.99
	Model only	0.74	0.54	**0.97**	0.54	0.95	0.87	0.77	0.94	0.94	0.90	0.90	0.93	0.90	0.92
	Both	**0.84**	**0.58**	**0.97**	**0.75**	**0.97**	0.87	**0.83**	**0.99**	**1.00**	**0.98**	**1.00**	**1.00**	**1.00**	**0.99**

same adoption rule (Appendix Table 3), adding record rankings, deletion fractions and actions. On Beers, Flights, Hospitals and Tax it adopts CARVEPrep's record-wise allocation, and these tables account for 4.20 of the 4.28 points. On Hospitals, adding only selection-then-repair loses 9 points, whereas the full set gains 11 and adopts a record-wise allocation instead of those repairs, so the value of a candidate depends on the rest of the menu, as expected when action effects need not add (Proposition E.6). Where the mean median witness-group size exceeds one, structural evidence recovers almost every injected violation, while label-conditional scoring recovers fewer than one in five (Appendix C.1). This is the gap that Proposition E.4 predicts for errors that leave the joint distribution of labels and predictions unchanged.

Feedback sharpens the priors, and held-out acceptance picks the better table. Across iterations, the updated prior ranks injected label errors higher on every table, raising its AUC (Figure 3(h)), and the loop raises post-training accuracy on Hospitals from 0.8533 to 0.9000 (Appendix Figure 6(f)). The held-out betting rule selects a preparation on 4 of the six tables. When Jellyfish's prepared table is added as a candidate, the rule selects it on Flights, where it is the stronger table, and still selects a CARVEPrep candidate on Hospitals and on every TEP scenario (Appendix Figure 6(g)), and CARVEPrep-Cert ranks second among the 21 non-oracle methods at 0.8010 (CARVEPrep-Cert row of Table 1).

Preparing the data changes what model tuning is worth. The last block of Table 1 tests joint data-model tuning. It crosses untouched and prepared data with fixed and reference-selected training rules, all from the same grid with the rule selected separately for each table. Tuning both achieves the highest or tied highest accuracy among the four configurations on 5 of 6 tables and on 6 of

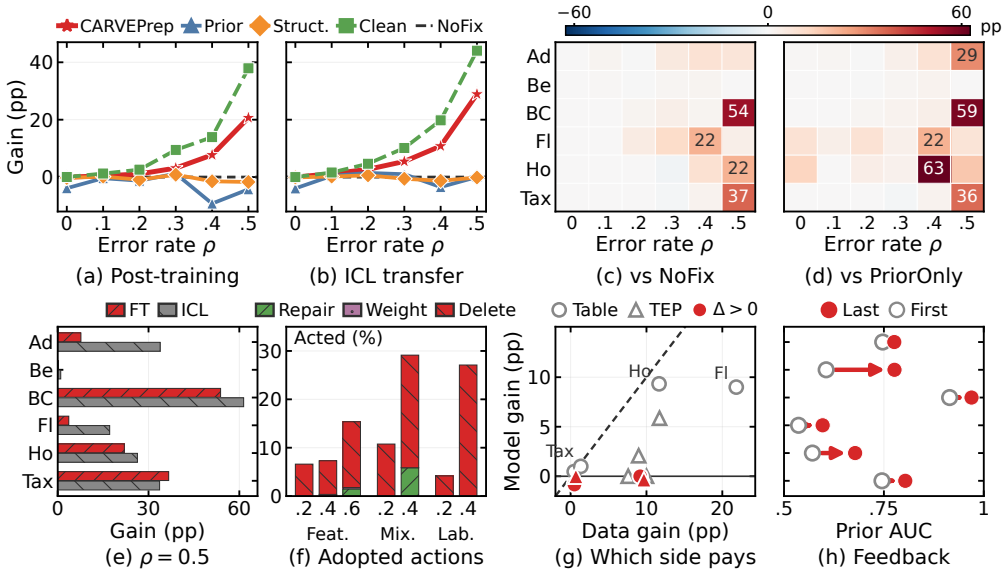


Figure 3: Error-rate response and preparation effects. (a,b) Six-table mean gains over NoFix under post-training and ICL, with **Prior** and **Struct.** for the two ablations. (c,d) Per-table post-training differences against NoFix and PriorOnly. (e) Per-table gains at $\rho = 0.5$ under post-training (**FT**) and ICL. (f) Pooled shares of repaired, downweighted (**Weight**) and deleted records by error type and rate. (g) Data-side against model-side gain on twelve tasks, with circles for tables, triangles for TEP scenarios, the diagonal indicating equal gains, and a filled marker for positive interaction. (h) First- and last-iteration prior AUC for injected label errors at $\rho = 0.4$ and the 0.3 record cap. **Feat.**, **Mix.** and **Lab.** denote feature, mixed and label errors, and dataset codes follow Table 8.

6 TEP scenarios. On Beers, training-rule selection alone changes accuracy by −0.84 points and preparation alone by +0.50, while the two together gain +2.83, an interaction of +3.17 points, as illustrated in Figure 1(b). The gain from selecting a training rule is negative before preparation and positive afterward, a local superadditive interaction between data preparation and model adaptation. Appendix C.3 compares the data-side and model-side gains task by task.

Mixing actions pays off beyond tables and beyond TabPFN. TEP has 52 sensor features and no declared dependencies. Allocation using sequence-specific evidence, the reference prior and the value model ranks first on five of six scenarios (Table 1). On Alpaca, the downstream model changes to the LLM Qwen2.5-1.5B-Instruct, and evidence comes from near-duplicate witnesses, format canonicalization and truncation detection. The adopted preparation still mixes actions, repairing 756, deleting 1,050 and downweighting 128 records on average, and reaches a held-out loss of 1.312, against 1.338 for an equal-size deletion by reference loss, 1.378 to 1.388 for the uniform strategies and 1.396 for NoFix (Appendix C.5).

5 Conclusion

CARVEPrep treats data preparation for a foundation model as an allocation problem that the model itself judges. Where changes land matters more than how many are made. Across six tables, adopted preparations average 0.8119, 2.66 points above the strongest baseline, which rewrites 70 times as many cells, while correct ground-truth repairs can still lower accuracy. The right action depends on the error type, structural evidence finds errors that predictions miss, and actions interact. A separable weighted least-squares analysis makes the budgeted gain of each action set exact and shows that dropping repair or fractional weights can strictly hurt. Data and model decisions are also coupled, since tuning both matches or beats either alone on 11 of 12 tasks and training-rule selection pays off on Beers only after preparation. With data-specific evidence, the same actions carry over to sensor time series and to LLM instruction tuning, and independent acceptance bounds the risk of adopting a worse table, even after retraining. Once a foundation model is post-trained on a task table, preparing that table belongs in the same loop as tuning the model.

Ethics statement

The experiments use public benchmarks with synthetic training-data errors and collect no new personal data. Adult contains demographic attributes. On Adult, the adopted preparation deletes and repairs at rates that are close across gender and race groups, and the group shares of the kept pool move by at most half a point (Appendix C.3, Table 5).

Reproducibility statement

https://anonymous.4open.science/r/CARVEPrep-ICLR2027/
Data and settings are in Appendix C.8, strategy contrasts in Appendix C.2, and proofs in Appendix E.

AI use statement

Generative AI tools were used to aid and polish the writing. Jellyfish-7B (Zhang et al., 2024) is used as a data-cleaning baseline, and Qwen2.5-1.5B-Instruct (Qwen et al., 2025) is the downstream model of the instruction-tuning experiment (Appendix C.5).

References

Mohamed Abdelaal, Christian Hammacher, and Harald Schöning. REIN: A comprehensive benchmark framework for data cleaning methods in ML pipelines. In *Proceedings 26th International Conference on Extending Database Technology, EDBT 2023, Ioannina, Greece, March 28-31, 2023*, pp. 499–511, 2023. doi: 10.48786/EDBT.2023.43.

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control. *CoRR*, abs/2110.01052, 2021. URL https://arxiv.org/abs/2110.01052.

Anastasios Nikolas Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*, 2024.

Stephen Bates, Anastasios Angelopoulos, Lihua Lei, Jitendra Malik, and Michael I. Jordan. Distribution-free, risk-controlling prediction sets. *J. ACM*, 68(6):43:1–43:34, 2021. doi: 10.1145/3478535.

Barry Becker and Ronny Kohavi. Adult. UCI Machine Learning Repository, 1996. URL https://doi.org/10.24432/C5XW20.

Laure Berti-Équille. Prior-aligned data cleaning for tabular foundation models. *CoRR*, abs/2604.25154, 2026. doi: 10.48550/ARXIV.2604.25154. URL https://arxiv.org/abs/2604.25154v1. Version 1, 28 April 2026.

Mengshi Chen, Yuxiang Sun, Tengchao Li, Jianwei Wang, Kai Wang, Xuemin Lin, Ying Zhang, and Wenjie Zhang. Empowering tabular data preparation with language models: Why and how? In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States, July 2-7, 2026*, pp. 228–246, 2026. doi: 10.18653/V1/2026.ACL-LONG.8.

James J Downs and Ernest F Vogel. A plant-wide industrial process control problem. *Computers & Chemical Engineering*, 17(3):245–255, 1993.

Amirata Ghorbani and James Y. Zou. Data shapley: Equitable valuation of data for machine learning. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97, pp. 2242–2251, 2019.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

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. *Nat.*, 637(8044):319–326, 2025. doi: 10.1038/S41586-024-08328-6.

Steven R Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, 2021.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*, 2022.

Johannes Jakubik, Michael Vössing, Niklas Kühl, Jannis Walk, and Gerhard Satzger. Data-centric artificial intelligence. *Bus. Inf. Syst. Eng.*, 66(4):507–515, 2024. doi: 10.1007/S12599-024-00857-8.

Mohammad Hossein Jarrahi, Ali Memariani, and Shion Guha. The principles of data-centric AI. *Commun. ACM*, 66(8):84–92, 2023. doi: 10.1145/3571724.

Leslie Kish. *Survey Sampling*. John Wiley & Sons, 1965. URL https://search.worldcat.org/title/Survey-sampling/oclc/256017.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70, pp. 1885–1894, 2017.

Sanjay Krishnan, Jiannan Wang, Eugene Wu, Michael J. Franklin, and Ken Goldberg. Activeclean: Interactive data cleaning for statistical modeling. *Proc. VLDB Endow.*, 9(12):948–959, 2016. doi: 10.14778/2994509.2994514.

Sanjay Krishnan, Michael J. Franklin, Ken Goldberg, and Eugene Wu. Boostclean: Automated error detection and repair for machine learning. *CoRR*, abs/1711.01299, 2017. URL http://arxiv.org/abs/1711.01299.

Yongchan Kwon and James Zou. Data-oob: Out-of-bag estimate as a simple and efficient data value. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202, pp. 18135–18152, 2023.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. From quantity to quality: Boosting LLM performance with self-guided data selection for instruction tuning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pp. 7602–7635, 2024. doi: 10.18653/V1/2024.NAACL-LONG.421.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, and Weizhu Chen. Not all tokens are what you need for pretraining. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.

Zifan Liu, Zhechun Zhou, and Theodoros Rekatsinas. Picket: guarding against corrupted data in tabular data during learning and inference. *VLDB J.*, 31(5):927–955, 2022. doi: 10.1007/S00778-021-00699-W.

Junwei Ma, Valentin Thomas, Rasa Hosseinzadeh, Alex Labach, 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 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025.

Mohammad Mahdavi and Ziawasch Abedjan. Baran: Effective error correction via a unified context representation and transfer learning. *Proc. VLDB Endow.*, 13(11):1948–1961, 2020.

Mohammad Mahdavi, Ziawasch Abedjan, Raul Castro Fernandez, Samuel Madden, Mourad Ouzzani, Michael Stonebraker, and Nan Tang. Raha: A configuration-free error detection system. In *Proceedings of the 2019 International Conference on Management of Data, SIGMOD Conference 2019, Amsterdam, The Netherlands, June 30 - July 5, 2019*, pp. 865–882, 2019. doi: 10.1145/3299869.3324956.

Sören Mindermann, Jan Markus Brauner, Muhammed Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N. Gomez, Adrien Morisot, Sebastian Farquhar, and Yarin Gal. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162, pp. 15630–15649, 2022.

Andrew Ng. A chat with Andrew on MLOps: From model-centric to data-centric AI, 2021. URL https://www.youtube.com/watch?v=06-AZXmwHjo.

Wei Ni, Xiaoye Miao, Xiangyu Zhao, Yangyang Wu, Shuwei Liang, and Jianwei Yin. Automatic data repair: Are we ready to deploy? *Proc. VLDB Endow.*, 17(10):2617–2630, 2024. doi: 10.14778/3675034.3675051.

Curtis G. Northcutt, Lu Jiang, and Isaac L. Chuang. Confident learning: Estimating uncertainty in dataset labels. *J. Artif. Intell. Res.*, 70:1373–1411, 2021. doi: 10.1613/JAIR.1.12125.

Vasileios Papastergios and Anastasios Gounaris. Out in the wild: Investigating the impact of imperfect data on a tabular foundation model. In *Proceedings of Workshops at the 51st International Conference on Very Large Data Bases, VLDB 2025, London, UK, September 1-5, 2025*, 2025.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 20596–20607, 2021.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. Tabicl: A tabular foundation model for in-context learning on large data. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267, 2025.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Theodoros Rekatsinas, Xu Chu, Ihab F. Ilyas, and Christopher Ré. Holoclean: Holistic data repairs with probabilistic inference. *Proc. VLDB Endow.*, 10(11):1190–1201, 2017. doi: 10.14778/3137628.3137631.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80, pp. 4334–4343, 2018.

Ivan Rubachev, Akim Kotelnikov, Nikolay Kartashev, and Artem Babenko. On finetuning tabular foundation models. *CoRR*, abs/2506.08982, 2025. doi: 10.48550/ARXIV.2506.08982. URL https://doi.org/10.48550/arXiv.2506.08982.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: An instruction-following llama model, 2023. URL https://github.com/tatsu-lab/stanford_alpaca.

Valentin Thomas, Junwei Ma, Rasa Hosseinzadeh, Keyvan Golestan, Guangwei Yu, Maksims Volkovs, and Anthony L. Caterini. Retrieval & fine-tuning for in-context tabular models. In

Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024, 2024.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J. Gordon. An empirical study of example forgetting during deep neural network learning. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, 2019.

Jean Ville. *Étude Critique de la Notion de Collectif*. Gauthier-Villars, 1939. URL https://catalogue.bnf.fr/ark:/12148/cb31580909m.

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(1):1–27, 2024.

Max Welling. Herding dynamical weights to learn. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, volume 382, pp. 1121–1128, 2009. doi: 10.1145/1553374.1553517.

William Wolberg. Breast Cancer Wisconsin (Original). UCI Machine Learning Repository, 1990. URL https://doi.org/10.24432/C5HP4Z.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. LESS: selecting influential data for targeted instruction tuning. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235, pp. 54104–54132, 2024.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. Data selection for language models via importance resampling. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

Daochen Zha, Zaid Pervaiz Bhat, Kwei-Herng Lai, Fan Yang, Zhimeng Jiang, Shaochen Zhong, and Xia Ben Hu. Data-centric artificial intelligence: A survey. *ACM Comput. Surv.*, 57(5):129:1–129:42, 2025. doi: 10.1145/3711118.

Haochen Zhang, Yuyang Dong, Chuan Xiao, and Masafumi Oyamada. Jellyfish: Instruction-tuning local large language models for data preprocessing. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pp. 8754–8782, 2024. doi: 10.18653/V1/2024.EMNLP-MAIN.497.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. LIMA: less is more for alignment. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

A Related Work

Table 2 summarizes the actions and information sources of the methods reviewed below using the definitions of this paper. Scores computed from label predictions alone are subject to the recall bound of Proposition E.4, while gradient-based scores additionally use model derivatives. CARVEPrep combines value changes and record weights with structural evidence, candidate evaluation and model feedback. The discussion follows the original methods. Table 2 identifies the components and uses being compared. Appendix C.8 specifies the experimental adaptations, including reference-based selection and Picket-style reconstruction.

General-task cleaning improves data correctness using constraints and statistical evidence. HoloClean combines integrity constraints with probabilistic inference (Rekatsinas et al., 2017). Raha learns error detectors from data signals with a few labeled tuples (Mahdavi et al., 2019), and Baran learns repair models (Mahdavi & Abedjan, 2020). Jellyfish instruction-tunes local language models

for preprocessing tasks including error detection and data imputation (Zhang et al., 2024). A recent survey organizes language-model-based tabular preparation by task and model usage (Chen et al., 2026).

Task-specific cleaning, selection and reweighting uses downstream performance to guide data decisions. ActiveClean links priority repair to statistical estimation (Krishnan et al., 2016), and BoostClean selects combinations from a library of detectors and repairs using validation performance (Krishnan et al., 2017). Record-selection scores include reducible held-out loss (Mindermann et al., 2022), early gradient norm (Paul et al., 2021), importance weights toward a target distribution (Xie et al., 2023), and estimates of a record's effect on validation loss (Koh & Liang, 2017; Xia et al., 2024). Data Shapley estimates marginal contributions to predictive performance, while Data-OOB reuses out-of-bag predictions to assign record values (Ghorbani & Zou, 2019; Kwon & Zou, 2023). Confident learning uses out-of-sample predicted probabilities and class-dependent confidence thresholds to identify and prune likely label errors (Northcutt et al., 2021), while meta-learned reweighting adjusts record weights through meta-gradients of clean-validation loss (Ren et al., 2018). For instruction tuning, IFD selects examples by instruction-following difficulty (Li et al., 2024), and LIMA uses a small set of carefully curated demonstrations (Zhou et al., 2023). During continued pretraining, Rho-1 selects token-level loss terms by excess loss relative to a reference model, while retaining the input context (Lin et al., 2024). Picket filters training records using reconstruction-loss statistics and detects corrupted inference queries with class-specific detectors using inputs and reconstruction losses (Liu et al., 2022).

Preparation for foundation models adapts task data to a pretrained model. Tabular foundation models predict from support records in context, or are post-trained on task-specific data (Hollmann et al., 2025; Qu et al., 2025; Ma et al., 2025). A systematic study of TabPFN-v2 compares full and parameter-efficient finetuning (Rubachev et al., 2025). It studies model adaptation and its effect on in-context prediction, and CARVEPrep adapts the task data. The tabular and time-series experiments use TabPFN-v2 as the downstream model, and the Alpaca experiment uses Qwen2.5-1.5B-Instruct. Motivated by prior alignment, L2C2 searches table-level operator sequences using downstream TFM accuracy, data-quality scores, row retention and a distributional-stability penalty relative to the original data (Berti-Équille, 2026). LoCalPFN retrieves nearest-neighbour support records for each query and fine-tunes the model on these local contexts (Thomas et al., 2024). QualiTab studies the effect of injected data errors on TabPFN representations and compares clean-context and corrupted-context scenarios (Papastergios & Gounaris, 2025). Definition B.1 specifies the operator class used to compare these methods.

Joint data-model optimization couples data decisions with model fitting. Among the task-specific methods above, ActiveClean (Krishnan et al., 2016) and BoostClean (Krishnan et al., 2017) tie cleaning decisions to model fitting. CARVEPrep repeatedly prepares data for the same pretrained downstream model and updates record priorities from context-held-out losses across iterations (Section 3.3).

Risk control and sequential testing give finite-sample guarantees for fixed rule families. Learn then Test provides multiple-testing procedures, including fixed-sequence testing, for risk control (Angelopoulos et al., 2021), and risk-controlling prediction sets use calibrated risk bounds (Bates et al., 2021). Under exchangeability and regularity conditions, Conformal Risk Control bounds the expected value of a bounded monotone loss (Angelopoulos et al., 2024). Betting confidence sequences and e-processes give anytime-valid bounds on bounded-observation means (Waudby-Smith & Ramdas, 2024; Howard et al., 2021). CARVEPrep applies betting to compare frozen data-model pairs on held-out records, and calibrates repair risk on an additional independent verification sample.

B Preliminaries

Objects and scope. This section fixes the objects the rest of the paper quantifies over. The definitions state the error classification that the evidence channels act on, the operator class whose outputs the four actions represent, the two coordinates that describe a prepared table, the two learning settings under which a preparation is scored and the witness rule that produces validated repair values. The assumption at the end is the sampling condition under which the acceptance guarantees of Appendix E.4 hold.

Table 2: Compared components and uses. Appr.: **G** general cleaning, **T** task-specific preparation, **F** foundation-model preparation, **DA** joint data-model optimization. Ops counts record-action categories only for the named cleaning or selection use, and selection need not permanently delete records. **user**, **gen.**, **ens.** and **seq.** denote user-specified cleaning, pseudo-sample generation, classifier ensembles and operator sequences, for which no fixed count is given. Sig.: **S** structural, **P** prior or label-conditional, **D** distributional information. Fit indicates integration with downstream model fitting, not whether the method itself learns a model.

Method	Appr.	Ops	Sig.	Fit
HoloClean (Rekatsinas et al., 2017)	G	2	S+D	✗
Raha and Baran (**Raha+Baran**) (Mahdavi et al., 2019; Mahdavi & Abedjan, 2020)	G	2	S+D	✗
Picket (Liu et al., 2022)	T	2	P+D	✓
ActiveClean (Krishnan et al., 2016)	T	user	P	✓
Cleanlab (label filtering) (Northcutt et al., 2021)	T	2	P	✓
Reducible holdout loss (**RHO-LOSS**), error L2-norm (**EL2N**) (Mindermann et al., 2022; Paul et al., 2021)	T	2	P	✓
Data selection with importance resampling (**DSIR**) (Xie et al., 2023)	T	2	D	✓
Influence-based subset selection (Koh & Liang, 2017; Xia et al., 2024)	T	2	P	✓
Out-of-bag data valuation (**Data-OOB**), forgetting, gradient norm (**GraNd**): subset selection (Kwon & Zou, 2023; Toneva et al., 2019; Paul et al., 2021)	T	2	P	✓
Herding (original generation method) (Welling, 2009)	T	gen.	D	✓
BoostClean (Krishnan et al., 2017)	T	ens.	S+P+D	✓
Jellyfish (detection/imputation) (Zhang et al., 2024)	F	2	P	✗
L2C2 (source of **PriorAlign**) (Berti-Équille, 2026)	F	seq.	P+D	✗
LoCalPFN (retrieval and fine-tuning) (Thomas et al., 2024)	F	2	P+D	✓
CARVEPrep	**DA**	**4**	**S+P**	✓

Definitions. Let D be a relational table with n records and d columns, n, d positive integers, where x_{ij} denotes the cell of record r_i in column j, X_j denotes column j, j_y the target column, and $\mathcal{V}(X_j)$ the legal value domain of column j. A set Σ of functional dependencies $A \to B$ states that records agreeing on column A must agree on column B. The decision unit throughout is the record and the evidence unit is the cell.

Let $\mathcal{N}_j$ be designated missing markers, disjoint from the legal domain $\mathcal{V}(X_j)$ containing the true cell value. An erroneous observation is missing if it belongs to $\mathcal{N}_j$, syntactic if it lies outside both sets, and otherwise an incorrect in-domain value. The last case is a semantic feature error when $j \neq j_y$ and label noise when $j = j_y$. This classification partitions erroneous cells, and a duplicate with correct values is handled by its weight. Missing-value indicators and format constraints provide cell-wise evidence. Marginal resampling redraws a cell from the clean marginal of its column, at a position selected independently of the clean values and independently of the record. It preserves that marginal in expectation while creating semantic errors, which structural evidence detects by comparing records agreeing on A (Definition B.5). Label noise requires evidence about the target column.

Definition B.1 (Preparation Operator)**.** A *preparation operator* maps D to (D', w), where D' has the same rows and columns as D and $w \in [0, 1]^n$. Each output row and weight depend on that record, its evidence and table statistics, but not on other records' outputs. Record creation and weights greater than one lie outside this class.

Coverage of the action representation. Appendix E.1 proves the canonical representation of admissible value-and-weight outputs and characterizes the search space under a record budget.

Definition B.2 (Quality State)**.** A *quality state* is a pair $\Psi(D_\pi, w) = (\kappa, \eta)$ of statistics of the prepared table and its weights. The coordinate κ is the *residual error*, the normalized weighted count of erroneous cells in the prepared table,

$$\kappa(D_\pi, w) \;=\; \frac{1}{nd} \sum_{i=1}^{n} w_i \big|\{j : x^{\pi}_{ij} \neq x^{\star}_{ij}\}\big|, \tag{3}$$

where $x^\star$ is the true value and x^π the prepared one and the sum runs over all d columns including the target. The coordinate η is the *effective coverage*, the effective sample size of the weights divided by the record count,

$$\eta(w) \;=\; \frac{1}{n} \cdot \frac{\left(\sum_i w_i\right)^2}{\sum_i w_i^2}, \tag{4}$$

which is the Kish effective sample size (Kish, 1965) divided by n for a nonzero weight vector, with $\eta(0) = 0$. Retaining m equally weighted records gives $\eta = \frac{m}{n}$, and unequal weights on the same retained set reduce it. DownweightAll assigns weight $w_{\text{min}} \in (0, 1)$ to every suspect record. A coordinate pair separates the four uniform strategies when NoFix, RepairAll, DeleteAll and DownweightAll map to four distinct points, which is the criterion used in Appendix C.7.

Definition B.3 (Settings)**.** In the *post-training setting*, require $\sum_i w_i > 0$. The loss on the surviving training records is

$$\bar{\ell} = \frac{\sum_i w_i \ell_i}{\sum_i w_i}.$$

Deleted records are removed from both training and context. Context construction samples the other records with base probabilities w_i and supplements the draw to maintain class coverage. In the *in-context setting*, the prepared table is used only as support, in which each positive-weight record appears once and zero-weight records are absent. ICL screening of candidates during selection instead samples the support set with inclusion probabilities w_i (Appendix D.3).

Definition B.4 (Reference Set)**.** The *reference set* D_{ref} is a small clean sample from the task, disjoint from the test set and shared by the compared methods. It represents the verified external information available in practice, namely trusted values, labels and expert rules. It supports prior fitting, candidate construction and evaluation, and early stopping, and it is excluded from the final downstream model's gradient-training data.

Definition B.5 (Witness and Evidence Tier)**.** Fix $A \to B$ in Σ and a record r_i. Its A-group is the set of records sharing its A value. The group is a *directed witness* for x_{iB}, written $x_{iB} \in \mathcal{W}_{\text{dir}}$, when the group holds at least m_{grp} records, a strict majority carries one B value, and x_{iB} is not that value. The group is an *undirected witness*, $x_{iB} \in \mathcal{W}_{\text{tie}}$, when it holds at least two distinct B values but no strict majority. The evidence category is $c_{ij} = 2$ for validated directed evidence, $c_{ij} = 1$ for unresolved suspicion and $c_{ij} = 0$ when no such evidence is present. An evidence-bearing cell has score $t_{ij} = c_{ij} + 0.999u_{ij}$, where $u_{ij} \in [0, 1]$ is its channel-specific strength, and cells without such evidence have $t_{ij} = 0$. These are the scores summed in the record ranking. A directed witness provides a majority candidate, which is used only when the majority share is at least r_{min} and the majority margin at least m_{min} (Appendix D.1). Proposition E.4 gives a lower bound on the probability that the true value is this majority and passes both thresholds. An undirected witness establishes a disagreement without providing a value, and a group of one record has no comparison for this dependency.

Example B.1 (Operators proposed as alternatives)**.** Subsampling at rate p assigns delete to the unsampled records and retain to the rest. Importance weighting normalized to $[0, 1]$ assigns downweight at strictly intermediate weights, retain at weight one and delete at weight zero. Imputation assigns repair with the imputed values. Label correction assigns repair restricted to the target column. Loss truncation assigns delete to the records above the loss quantile. Curriculum ordering is a schedule over w, which gives a sequence of allocations and adds no action type. Each is an allocation of the four action types of Definition 2.2, defined on the suspect set when the operator acts only there and on the whole table otherwise.

The population guarantees of Appendix E.4 hold only for a sample drawn independently of everything used to build and select a preparation, which the following assumption states.

Assumption B.1 (Independent evaluation sample)**.** The sample used for a population guarantee consists of independent, identically distributed clean records from the task population. Conditional on all development data, the candidate policies, trained evaluation models, the reference data-model pair used for comparison and the score functions are fixed before this sample is accessed. Fitting, early stopping and calibration use separate data. Each acceptance record is revealed once in a fixed order, and bets use only previously revealed records and development information.

C Experiments

Effects of preparation and model fitting. Figures 4, 5 and 6 present the gains across tasks, the strategy contrasts and the interaction between preparation and model fitting. We then examine scale and transfer to other data types, specify the motivating comparisons and quality-state screening, and give the full protocol in Appendix C.8.

C.1 Generalization and Evidence Comparisons

Preparation gains across tasks. Figure 4 shows the distribution of gains across the tabular datasets and the TEP scenarios, supplementing the averages in Table 1. The comparisons below then examine robustness to error rate, the contribution of the candidate set and the information supplied by each evidence channel.

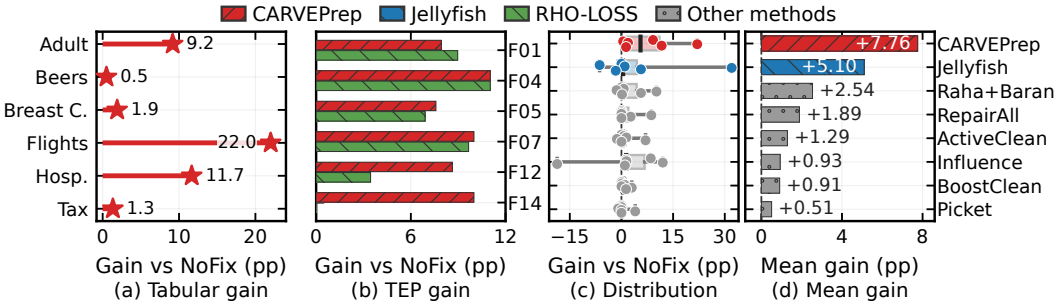


Figure 4: Gains over NoFix in the main comparison. (a) CARVEPrep on each tabular dataset. (b) CARVEPrep and RHO-LOSS on the TEP scenarios. (c,d) Distribution and mean of the six table-wise gains for CARVEPrep and the seven other methods with the highest tabular mean, with boxes at the quartiles, whiskers at the minimum and maximum and one point per table.

Error-rate response. The error-rate study of Figure 3(a–e) covers six tables, six mixed-error rates from 0 to 0.5, five preparations and the post-training and ICL settings, giving 360 measurements. A further 100 measurements cover positive label-error rates on Beers and Breast Cancer. Preparations are selected for post-training and then also evaluated through ICL. For the five positive label-error rates, post-training gains over NoFix range from 0.00 to 1.00 points on Beers and from 0.00 to 60.00 on Breast Cancer. The largest Breast Cancer gain occurs at rate 0.5, where NoFix has balanced accuracy near chance.

Candidate-set contribution. The nested sets compare complete candidate-construction and search procedures. C adds record rankings and cleaning fractions as well as action combinations. Under the main comparison setting, candidate set A contains NoFix and the Ref. likelihood and RHO-LOSS deletion candidates, B adds the two selection-followed-by-repair candidates to A, and C contains the complete candidate set. They share candidate reference scores and the adoption rule. Mean accuracies are 0.7710 for A, 0.7702 for B and 0.8139 for C. On Hospitals, B adopts RHO-LOSS selection followed by repair and falls 9 points below A. With the complete set, Beers, Flights, Hospitals and Tax adopt CARVEPrep's record-wise allocation, Adult adopts RHO-LOSS selection followed by repair, and Breast Cancer keeps the Ref. likelihood candidate of A.

Evidence comparison. Four scores evaluate the same injected violations on 21 dependencies from four tables at a fixed record budget, namely label-conditional loss, column-conditional scores with the record included in or held out of the context, and structural evidence. Across the 18 dependencies with median group size above one on average over seeds, structural recall is 0.947, with a minimum of 0.827 across dependencies. Label-conditional recall is 0.184, with a maximum of 0.440. The two column-conditional scores achieve 0.579 and 0.487, respectively. On the three dependencies with median group size one on average over seeds, structural and label-conditional recall are 0.271 and 0.142. The structural advantage increases with median group size (Spearman correlation 0.416 across the 21 dependencies), consistent with the group-size dependence in Proposition E.4. Conditional entropy is zero for 20 dependencies and 0.040 for the remaining one, so these dependencies are exact

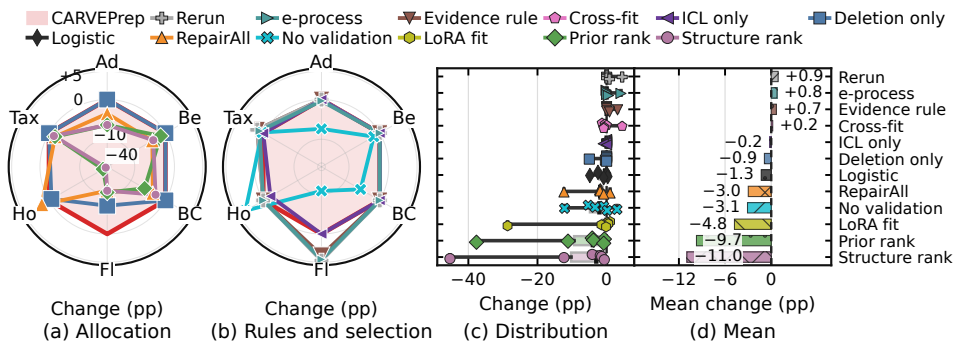


Figure 5: Strategy contrasts on six tables, as changes relative to CARVEPrep with positive values favoring the contrast. (a,b) Radial axes in $\operatorname{asinh}(\frac{\Delta}{2})$ with percentage-point ticks, and the shaded area marks the zero change of CARVEPrep. (c) Per-table changes with min–max whiskers. (d) Six-table means. Dataset codes follow Figure 3.

or nearly exact, which is the condition under which Proposition E.4 bounds the failure probability of the group majority.

C.2 Strategy Contrasts

Allocation and validation in complete procedures. Figure 5 contrasts twelve procedures with the full method on six tables. Rerun repeats the full method to show training variation. Since actions interact (Proposition E.6), these differences compare procedures, rather than additive contributions of individual steps. Under such interaction, the effect of removing one component depends on the rest of the procedure: a repair that is worthless alone can be the step that makes a deletion useful, and a gain measured in isolation need not survive combination with other actions (Figure 1(b), Example C.2). A leave-one-out difference therefore measures a component's marginal effect inside the adopted preparation, not its contribution to the joint gain, so these comparisons read whole procedures and nested candidate sets under one adoption rule.

The allocation procedures in panel (a) are defined as follows. Deletion only keeps the deletions of the policy selected by the full method, reverses its value changes and sets its positive weights to one. RepairAll applies all validated candidates. Prior rank and Structure rank order records by one signal and assign the four actions by a fixed rule, without learned action values or candidate acceptance. Under the same record cap, the separate PriorOnly ablation deletes prior-ranked records, and StructuralOnly applies validated dependency repairs and deletes records with unresolved violations.

The rule and selection procedures in panel (b) are defined as follows. No validation applies all available structural candidates with unit weights, without the validation condition, the allocation budget or candidate selection. Evidence rule replaces the action map inside the candidate-selection procedure. ICL only selects candidates by their ICL scores without the short post-training runs. e-process runs the betting test of Proposition E.7 on the development reference set in place of a separate acceptance sample. Of the procedures shown only in panels (c,d), Rerun repeats the full method, Logistic replaces the candidate scorer with a logistic model, LoRA fit replaces final training, and Cross-fit replaces the reference-fitted prior and its label-contradiction branch while other candidates still read reference labels.

Averaged over repeated training on four tables, Deletion only and RepairAll are 0.007 and 0.047 below the full method, and the better of Prior rank and Structure rank is 0.122 below it. The gap of Deletion only to the full method estimates $\Delta(\sigma\tau) - \Delta(\sigma)$ of Appendix E.3.2 with accuracy in place of risk, where σ denotes the adopted deletions and τ the value and weight changes. No validation costs 5.3 points on average and has repair error rate 0.229. LoRA fit reaches 99% of the full-fit accuracy with 2.5% trainable parameters.

C.3 Action selection, training-rule selection and held-out acceptance

Variation, interaction and held-out evaluation. Figure 6 compares repeated training, action sets, selection variants, training rules, feedback and held-out acceptance. The error-rate and candidate-set

Table 3: Candidate-set comparison at $\rho = 0.4$ in final-model accuracy. A contains NoFix and two selection candidates; C uses the full candidate set, with changes in percentage points. Bold shaded cells mark the largest accuracy in each row.

Dataset	A Restricted	C Full set	Δ(C–A)
Adult	0.8300	**0.8350**	+0.50
Beers	0.5533	**0.5550**	+0.17
Breast Cancer	**0.9714**	**0.9714**	+0.00
Flights	0.5500	**0.6667**	+11.67
Hospitals	0.8633	**0.9767**	+11.34
Tax	0.8583	**0.8783**	+2.00
Mean	0.7710	**0.8139**	+4.28

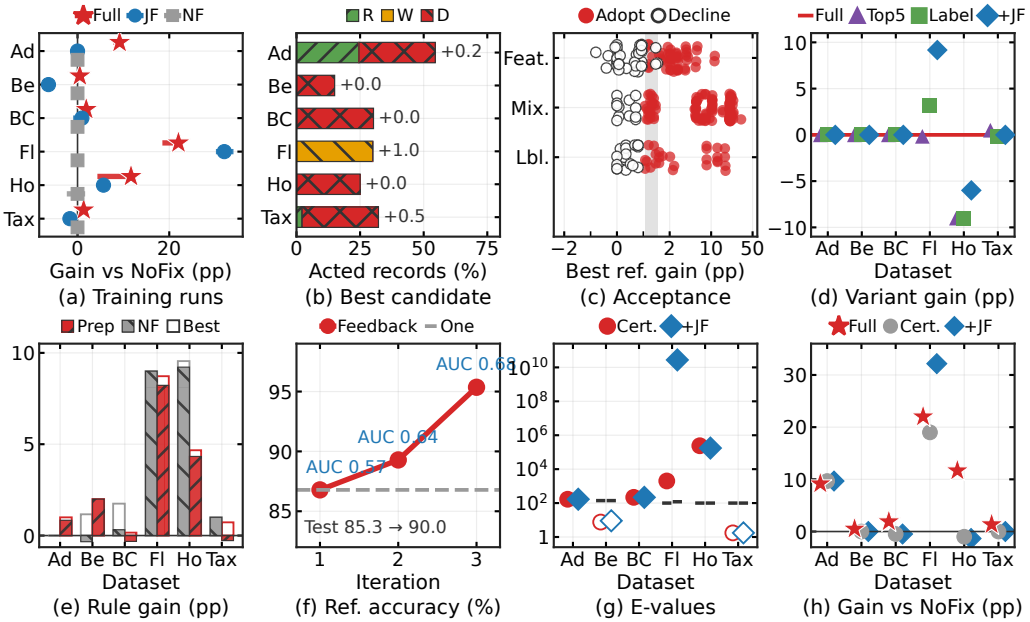


Figure 6: Studies at $\rho = 0.4$, with **Full**, **JF** and **NF** for CARVEPrep, Jellyfish and NoFix. (a) Gain over NoFix, with bars over three training runs and markers at the values of Table 1. (b) Actions of the test-best candidate under feature errors, repair (**R**), downweight (**W**) and delete (**D**), with its gain over the best retain-and-delete candidate. (c) Adoption traces, with the band between the largest declined and the smallest adopted reference gain. (d) Changes from the default configuration with five post-trained candidates (**Top5**), a label check after repair (**Label**) or the Jellyfish table as a candidate (**+JF**). (e) Gains of the rule selected on the reference set (filled) and of the test-best rule (outline) on the prepared (**Prep**) and the untouched table. (f) Hospitals feedback at a 0.3 record cap against one preparation (**One**), annotated with prior AUC and final test accuracy. (g) Maximum betting statistics of held-out acceptance (**Cert.**) with and without the Jellyfish table, with black lines at the number of tested candidates divided by δ_2 and open markers for no adoption. (h) Gains over NoFix with bootstrap adoption and the two held-out acceptance configurations. Dataset codes follow Figure 3.

studies of the main text use the splits and injections of the main comparison. At the main comparison's error rate, their six-table means lie within 0.5 points of it, CARVEPrep at 0.8139 and NoFix at 0.7368 against 0.8119 and 0.7343. Table 3 gives the comparison of A and C by table, and Figure 6 collects the studies of this subsection.

Preparation and training-rule comparisons. The last block of Table 1 and the right block of Table 6 take all four configurations of a comparison from the grid described below. On easy tasks, dozens of runs share the highest reference accuracy. We break these ties by preferring the fixed training rule, followed by a fixed order of the remaining rules that is released with the code and uses

Table 4: Main comparison over three training runs. Every cell shows the run selected by reference accuracy and, in brackets, the lowest and highest test accuracy over the three training runs of that cell. The last two columns give the six-table mean of the selected runs and of all runs.

Method	Adult	Beers	Breast	Flights	Hosp.	Tax	Mean selected	Mean all runs
NoFix	0.74 [0.74, 0.74]	0.55 [0.54, 0.55]	0.95 [0.95, 0.96]	0.45 [0.45, 0.45]	0.86 [0.83, 0.87]	0.86 [0.86, 0.86]	0.73	0.73
RepairAll	0.74 [0.74, 0.74]	0.55 [0.54, 0.55]	0.95 [0.95, 0.96]	0.54 [0.45, 0.54]	0.88 [0.87, 0.88]	0.86 [0.86, 0.86]	0.75	0.74
DeleteAll	0.74 [0.74, 0.74]	0.55 [0.53, 0.55]	0.95 [0.93, 0.95]	0.43 [0.43, 0.43]	0.31 [0.31, 0.31]	0.50 [0.50, 0.50]	0.58	0.58
DownweightAll	0.74 [0.74, 0.74]	0.54 [0.54, 0.54]	0.96 [0.92, 0.96]	0.44 [0.44, 0.45]	0.51 [0.28, 0.51]	0.86 [0.86, 0.86]	0.67	0.65
RandomDelete	0.75 [0.74, 0.75]	0.54 [0.54, 0.54]	0.83 [0.70, 0.83]	0.51 [0.45, 0.51]	0.28 [0.28, 0.28]	0.87 [0.87, 0.87]	0.63	0.61
ActiveClean	0.74 [0.74, 0.75]	0.55 [0.54, 0.55]	0.97 [0.91, 0.97]	0.44 [0.44, 0.45]	0.93 [0.84, 0.93]	0.86 [0.86, 0.86]	0.75	0.73
Picket	0.74 [0.74, 0.74]	0.55 [0.54, 0.55]	0.94 [0.90, 0.94]	0.45 [0.45, 0.45]	0.90 [0.87, 0.90]	0.86 [0.86, 0.86]	0.74	0.73
Cleanlab	0.71 [0.71, 0.71]	0.54 [0.54, 0.54]	0.90 [0.88, 0.90]	0.56 [0.55, 0.56]	0.84 [0.53, 0.84]	0.73 [0.72, 0.73]	0.71	0.68
PriorAlign	0.74 [0.74, 0.74]	0.54 [0.54, 0.54]	0.95 [0.95, 0.96]	0.45 [0.45, 0.45]	0.84 [0.78, 0.84]	0.86 [0.86, 0.86]	0.73	0.73
Raha+Baran	0.74 [0.74, 0.74]	0.55 [0.54, 0.55]	0.94 [0.91, 0.94]	0.55 [0.54, 0.55]	0.91 [0.84, 0.91]	0.86 [0.77, 0.86]	0.76	0.74
BoostClean	0.74 [0.74, 0.74]	0.55 [0.54, 0.55]	0.96 [0.91, 0.96]	0.45 [0.45, 0.45]	0.89 [0.28, 0.89]	0.87 [0.79, 0.87]	0.74	0.69
Jellyfish	0.74 [0.74, 0.74]	0.48 [0.48, 0.49]	0.96 [0.94, 0.96]	0.77 [0.75, 0.79]	0.91 [0.91, 0.91]	0.84 [0.84, 0.86]	0.79	0.78
DSIR	0.74 [0.74, 0.74]	0.55 [0.54, 0.55]	0.95 [0.94, 0.95]	0.45 [0.45, 0.45]	0.77 [0.74, 0.77]	0.86 [0.83, 0.86]	0.72	0.71
EL2N	0.37 [0.31, 0.37]	0.23 [0.23, 0.23]	0.50 [0.15, 0.50]	0.41 [0.33, 0.41]	0.31 [0.30, 0.31]	0.46 [0.43, 0.46]	0.38	0.33
RHO-LOSS	0.82 [0.82, 0.84]	0.54 [0.54, 0.55]	0.97 [0.96, 0.97]	0.59 [0.58, 0.59]	0.43 [0.43, 0.45]	0.80 [0.78, 0.80]	0.69	0.69
Ref. likelihood	0.83 [0.82, 0.83]	0.56 [0.55, 0.56]	0.97 [0.97, 0.98]	0.57 [0.55, 0.57]	0.67 [0.43, 0.67]	0.87 [0.86, 0.87]	0.74	0.72
Data-OOB	0.76 [0.76, 0.79]	0.54 [0.47, 0.54]	0.89 [0.88, 0.89]	0.49 [0.46, 0.49]	0.68 [0.38, 0.68]	0.79 [0.76, 0.79]	0.69	0.65
Herding	0.74 [0.74, 0.74]	0.55 [0.54, 0.55]	0.94 [0.83, 0.94]	0.49 [0.45, 0.49]	0.28 [0.28, 0.28]	0.87 [0.87, 0.87]	0.64	0.63
GraNd	0.69 [0.63, 0.69]	0.54 [0.51, 0.54]	0.94 [0.91, 0.94]	0.58 [0.47, 0.58]	0.74 [0.44, 0.74]	0.83 [0.82, 0.83]	0.72	0.67
CARVEPrep	0.84 [0.82, 0.84]	0.55 [0.55, 0.56]	0.97 [0.97, 0.98]	0.67 [0.63, 0.67]	0.97 [0.90, 0.97]	0.87 [0.87, 0.88]	0.81	0.80
Clean (oracle)	0.85 [0.85, 0.85]	0.57 [0.57, 0.57]	0.97 [0.97, 0.98]	1.00 [1.00, 1.00]	0.99 [0.99, 1.00]	0.87 [0.87, 0.87]	0.88	0.88

no test data. These results therefore differ by a fraction of a point from the cells of Table 1 and from the left block of Table 6, which report separate runs for the method comparison.

Variation across training runs. Ties in reference accuracy follow the fixed order of the three runs. Averaged over three training runs of every cell (Figure 6(a)), CARVEPrep reaches 0.8028 against 0.7840 for the strongest baseline and 0.7333 for NoFix. Table 4 gives each selected run and range under the same split and corruption instance.

Selected actions. In the main comparison of Table 1, each table adopts one candidate. Adult adopts RHO-LOSS deletion followed by verified repair of the retained records, which repairs 253 records, deletes 270 and changes 306 cells at repair error rate 0.0817, against 419 cells at 0.0907 for RepairAll and 6,868 cells at 0.4520 for Jellyfish. Beers adopts the record-wise allocation at fraction 0.15, deleting 135 records, Flights the one at fraction 0.2, deleting 180, Hospitals and Tax the ones at fraction 0.25, deleting 105 and 225, and Breast Cancer the Ref. likelihood candidate on 89 records. Adult's complete candidate affects 523 of 900 training records. Its deletion stage obeys the 0.3 record cap, and the verified repairs then apply to retained records. The full system compares this complete preparation with budgeted allocations under the same adoption rule.

Group statistics of the adopted Adult preparation. The injected corruption is independent of group membership. The adopted Adult policy deletes 30.0% and repairs 28.1% of the pool, at rates that are close across the gender and race groups (Table 5). Deletion rates span 26.1% to 32.3%, and every group's share of the retained pool moves by at most half a point from its share of the original pool. Black records are deleted least often and repaired most often, and their share of the retained pool rises from 9.8% to 10.3%.

Best evaluated candidates by action set. Under feature errors at the main comparison's error rate, we post-train each candidate and compare test accuracy retrospectively. The best evaluated candidate of the full action set exceeds the best evaluated retain-and-delete candidate on three of six tables. It is prior-guided downweighting on Flights and RHO-LOSS selection followed by verified repair on Adult and Tax (Figure 6(b)).

Operating threshold of the adoption rule. Figure 6(c) plots the reference gain and the outcome of every adoption decision reported in Section 4.2. Adopted and declined gains overlap in a narrow band because the paired bootstrap depends on the distribution of per-record differences as well as on the mean gain.

Table 5: Group statistics of the adopted Adult preparation. Deletion and repair rates are computed within each group, and Kept share is the group's share of the 630 retained records. The adopted policy deletes 270 and repairs 253 of the 900 training records.

Group	Records	Share (%)	Deleted (%)	Repaired (%)	Kept share (%)
Female	291	32.3	29.2	28.5	32.7
Male	609	67.7	30.4	27.9	67.3
White	781	86.8	30.3	27.0	86.3
Black	88	9.8	26.1	37.5	10.3
Other	31	3.4	32.3	29.0	3.3

Variants of the default configuration. Figure 6(d) compares three variants with the default configuration on the main comparison, each selected on the reference set. Restricting post-training to five candidates, and adding a label test after repair, each lower the six-table mean. Admitting Jellyfish's prepared table into the candidate set results in its adoption on two of the six tables. On Flights the adopted table reaches 0.7600, +9.17 points over the default configuration, after 3,039 rewritten cells of which 67% differ from the truth. On Hospitals its reference accuracy is higher by 1.9 points, but its test accuracy changes by -6.00 points after 4,111 modified cells at 44% error. This discrepancy motivates the held-out comparison below.

Selecting the training rule on the prepared table. The comparisons above vary the preparation under one fixed training rule. This comparison varies the training rule as well, on the untouched and on the prepared table, which separates the two decisions on the same tasks. The grid crosses epoch caps $\{5, 10, 20, 40\}$, learning rates $\{10^{-5}, 3 \times 10^{-5}, 10^{-4}\}$, full parameters or the LoRA adapter, and one or four estimators, 48 rules in all. Every rule trains on the untouched table and on the CARVEPrep table of the main comparison, the reference accuracy selects the rule, and the test accuracy of the selected rule is reported (last block of Table 1, Figure 3(g)). Preparation alone gains +7.50 points over the untouched table under the fixed training rule, rule selection alone +3.16, and both together +9.27. The rule selected for the prepared table differs from the rule selected for the untouched table on every table. On Adult and Beers, reference accuracy selects the adapter for the prepared table and full parameters for the untouched table. On the untouched Adult table, none of the training rules in the grid improves accuracy or recovers the loss due to corrupted records.

Selecting the training rule on prepared data never falls below training-rule selection alone, with one tie on Breast Cancer, and improves accuracy by +4.14 to +10.00 points on the TEP scenarios. The interaction on TEP scenario F01 is +3.45 points. On the 1,250-record Adult pool, training-rule selection alone yields no improvement, while the same search contributes +3.73 points once the table is prepared (Appendix C.4).

Gains in Figure 6(e) are relative to the three-run mean under the fixed training rule. Selection on the reference set recovers most of the available gain in the grid. On the untouched table it gains +3.20 points over the fixed training rule where the best rule of the grid on the test split offers +3.74, and on the prepared table +2.47 against +2.88, capturing 85% and 86% of the largest gain in the grid (Figure 6(e)). When comparison on the reference set is limited to k rules, the fixed training rule always among them and the others drawn from the grid, the prepared table gains +0.84 points at four rules and +1.37 at twelve against +1.78 for the full grid. With at least 6 rules, the untouched table gains more from the same search, +3.00 points at the full grid. Four rules cost 7% of the full grid's training time for 47% of its gain.

Differences across datasets. Whether preparation or training-rule selection yields the larger gain varies across the tasks, and on some tasks the combined gain exceeds the sum of the two. Preparation gains exceed training-rule gains on Hospitals (+11.66 against +9.33 points), Flights (+21.84 against +9.00) and Adult. On Tax, the original table is within 1.17 points of clean training, and neither change gains more than +1.34 points. On the 5,000-record Tax pool the rule retains NoFix, and training-rule selection gains +0.67 points (Appendix C.4). Beers, the small Adult pools and F01 benefit from both changes and show a positive interaction.

Feedback loop. The loop of Algorithm 3 re-ranks records by the losses of the post-trained model, and the measurement is whether the prior and the adopted candidate change across rounds. The loop

of Algorithm 3 runs three rounds at the 0.3 record cap. Each round after the first updates the prior from the record losses of the previous round and compares the retained table with new candidates using the same training and adoption rules (Appendix D.4). The AUC of the prior against the injected label-noise mask rises on every table, both at the fixed cap (Figure 3(h)) and with the fraction grid of the main comparison, where a frozen-prior control that computes the record losses without updating the prior leaves the AUC unchanged on every table and a shuffled-prior control that permutes the losses across records lowers it on every table. On Hospitals at the fixed cap, the reference accuracy of the adopted candidate rises in every round and the final model goes from 0.8533 with one preparation to 0.9000 with the loop (Figure 6(f)).

Acceptance on a held-out sample. Selection on the whole reference set gives no bound on the population score of the adopted candidate. Figure 6(g,h) evaluates the score-acceptance procedure of Proposition E.7. The reference set is split once into 60% development and 40% acceptance records, which are used as in Appendix D.5. The betting e-process reads the h acceptance records in a fixed order with $\epsilon = 0$ and $\delta_2 = 0.05$ and adopts a candidate only when its maximum betting statistic $\max_{t \le h} E_{k,t}$ reaches $\frac{K_c}{\delta_2}$, where K_c is the number of tested candidates. The rule selects a preparation on 4 of six tables, CARVEPrep's record-wise allocation on Breast Cancer and a selection-followed-by-repair candidate on Adult, Flights and Hospitals, and retains NoFix on Beers and Tax. Against bootstrap adoption on the whole reference set held-out acceptance costs 3.17 points through the smaller development set and 0.06 points through the e-process itself, measured against the bootstrap rule under the same split. With Jellyfish's prepared table in the candidate set, the acceptance rule selects it on Flights with a maximum betting statistic of 2.8×10^{10}, +13.17 points over held-out acceptance without it, and does not select it on Hospitals. There, its development accuracy is higher by +6.55 points, but it ranks below the RHO-LOSS selection-followed-by-repair candidate on the acceptance sample (maximum betting statistics 175,099 against 10,097, acceptance gains 0.265 against 0.241). The other tables adopt the same policy with and without the external table. On the six TEP scenarios the acceptance rule never selects the external table, whose own accuracy is lower on every scenario, and selects one of CARVEPrep's deletion candidates on each (CARVEPrep-Cert row of Table 1). The population bound of Proposition E.7 applies to the frozen evaluation model under the sampling conditions of Assumption B.1, and Corollary E.2 transfers it to the retrained model through a prediction-disagreement bound on fresh unlabeled records.

C.4 Scale robustness

More training records under a fixed reference cap. Table 6 expands Adult and Tax from 1,250 to 10,000 records, with Raha+Baran and clean training at every scale. Pool sizes include training, reference and test records. The main Adult pool has 900 training, 500 reference and 600 test records, and the 10,000-record pool has 6,500, 500 and 3,000, respectively. Thus the reference-to-training ratio falls from 55.6% to 7.7%, while the reference share of the full pool falls from 25% to 5% (28% at the smallest pool). The common main-study pool cap controls the cost of comparing 22 methods, while the scale study holds the 500-record reference cap fixed as training grows. It tests how preparation changes as training grows under a fixed reference cap. On Adult, CARVEPrep stays above NoFix at every pool, by 1.7 to 9.2 points, and above Raha+Baran at every pool. On Tax, CARVEPrep, NoFix and Raha+Baran all lie within 1.1 points of clean training at every pool, leaving little potential gain from preparation, while Jellyfish falls 5.9 points behind clean training on the smallest pool.

The size of the reference set limits how small a gain the comparison can detect. With a few hundred reference records the candidates remain separable on pools up to several thousand records, and on the 10,000-record Adult pool both rules move toward the untouched table. The bootstrap rule adopts the smallest cleaning fraction and reaches 0.8060, against 0.8120 for Jellyfish and 0.7890 for NoFix, and the held-out rule with Jellyfish's table in the candidate set adopts no candidate and reaches 0.775 to 0.792.

Smaller reference caps. Under feature errors, reference caps of 125, 250 and 500 yield six-table reference-selected accuracies of 0.8543, 0.8660 and 0.8505, and their three-training-run means are 0.8544, 0.8616 and 0.8480. Adult reaches 0.8283, 0.8500 and 0.8150. Reducing the reference cap returns records to training, so these runs vary both the verified information and the noisy training pool.

Table 6: Scale study across pool sizes. The left block compares methods on each pool. The right block compares preparation and training-rule selection, as in the last block of Table 1, with Δ denoting their interaction in percentage points. Bold shaded cells mark the best non-oracle value in each row and block. The 2,000-record Adult row repeats Table 1.

		Methods on the pool					Two sides of the decision				
Table	Pool	NoFix	Raha+Baran	Jellyfish	CARVEPrep	Clean	Untouched	Data only	Model only	Both	Δ
Adult	1,250	0.7493	0.7440	0.7387	**0.8027**	0.8160	0.7333	0.8027	0.7333	**0.8400**	+3.73
Adult	2,000	0.7433	0.7433	0.7433	**0.8350**	0.8483	0.7433	0.8350	0.7433	**0.8367**	+0.17
Adult	2,500	0.7973	0.7453	0.7453	**0.8333**	0.8573	0.7973	0.8360	0.7973	**0.8387**	+0.27
Adult	5,000	0.7420	0.7420	0.7827	**0.8320**	0.8533	0.7420	**0.8313**	0.7420	0.8227	−0.86
Adult	7,000	0.7990	0.7476	0.8090	**0.8386**	0.8610	0.7976	**0.8419**	0.7976	**0.8419**	+0.00
Adult	10,000	0.7890	0.7523	**0.8120**	0.8060	0.8610	0.7893	0.8050	0.7893	**0.8083**	+0.33
Tax	1,250	0.8533	0.8613	0.8000	**0.8667**	0.8587	0.8533	**0.8667**	0.8613	**0.8667**	−0.80
Tax	2,500	**0.8747**	0.8733	0.8667	0.8693	0.8800	**0.8760**	0.8707	0.8707	0.8707	+0.53
Tax	5,000	0.8580	**0.8587**	0.8560	0.8580	0.8680	0.8580	0.8580	**0.8647**	0.8607	−0.40
Tax	7,000	0.8762	0.8762	**0.8805**	0.8719	0.8795	0.8762	0.8738	**0.8795**	0.8748	−0.23
Tax	10,000	0.8630	**0.8743**	0.8737	0.8620	0.8647	**0.8643**	0.8630	0.8627	0.8617	+0.03

Recorded computation at the largest pool. At 10,000 Adult records, preparation, final training and scoring take means of 177.1 and 217.5 minutes in two recorded CARVEPrep batches, of which candidate post-training accounts for 147.6 and 197.5 minutes. The first batch records 4.6 minutes for NoFix and 10.4 for Jellyfish. These timings exclude model loading, splitting and error injection. Hardware identity was not recorded, so they describe observed search cost rather than a controlled speed comparison.

C.5 Operator Extension Beyond Tables

The operator representation transfers across data types. On sequential sensor tables and on instruction and response records, the four actions, the value-and-weight representation and the comparison of complete preparations are unchanged (Table 7). Evidence channels and record priors are chosen for each data type. For text, allocation uses an evidence-based rule in place of the learned action-value model. It screens complete candidates by the weighted mean base-model response loss of retained records, then post-trains the selected candidates with LoRA and compares their reference losses. The extension therefore shares the operator class and the evaluation of complete preparations, with different allocation and scoring rules.

Table 7: Operator extension across data modalities. Every setting uses retain, repair, delete and downweight, represented by record values and weights, and evaluates complete preparations.

Component	Relational tables	Sensor sequences	Instruction text
Evidence	Dependency witnesses, format and typo checks	Sequence spike checks; no declared dependencies	Near-duplicate witnesses, format canonicalization, truncation detection
Record prior	Reference-conditional label score	Reference-conditional label score	Base-model response loss given the instruction
Allocation	Learned action values	Learned action values	Evidence-based rule
Candidate scoring	ICL screening, then short post-training accuracy	ICL screening, then short post-training accuracy	Weighted base-model response loss, then LoRA reference loss
Downstream model	TabPFN-v2	TabPFN-v2	Qwen2.5-1.5B-Instruct
Reported metric	Test accuracy	Test accuracy	Held-out response loss

On the six TEP fault scenarios, the 52-feature sensor table has no declared dependencies and hence no relational dependency witnesses. Allocation uses sequence-specific spike checks, the reference prior and the value model. CARVEPrep ranks first on five of the six scenarios and outperforms RHO-LOSS and NoFix (Table 1, Appendix Figure 4(b)). The comparison uses the original fault01 scenario and faults 04, 05, 07, 12 and 14, and the fitting and test portions follow sequence order. Scenarios 02 and 08, not part of the comparison above, behave similarly. CARVEPrep is close to

NoFix on scenario 02 (0.9034 against 0.9069) and leads it on scenario 08 (0.9241 against 0.8862), with the clean-training reference at 0.9586 and 0.9414 respectively.

The instruction-tuning experiment below changes the record itself and not only the evidence channels. In the Alpaca (Taori et al., 2023) supervised post-training experiment, each record is an instruction and response pair. Evidence comprises near-duplicate witnesses, lossless format canonicalization and truncation detection, and the prior is the base model's response loss conditional on the instruction. The downstream model is Qwen2.5-1.5B-Instruct (Qwen et al., 2025) post-trained with LoRA (Hu et al., 2022) on 3,500 training records, with 200 reference and 500 test records held out from a pool of 5,000. The mean held-out loss is 1.312, against 1.338 for the best deletion baseline and 1.396 for NoFix. The adopted preparation repairs 756, deletes 1,050 and downweights 128 records on average. Repair and downweight counts can overlap when a repaired record receives a reduced weight. The uniform strategies DeleteAll, RepairAll and DownweightAll reach mean losses of 1.378, 1.387 and 1.388, so repair alone and downweighting alone each lower the mean NoFix loss by less than 0.01. The best deletion baseline ranks records by reference loss and removes 1,050 of them, the number deleted by the adopted preparation under a different ranking. The adopted preparation reaches a lower loss on each of the three runs, by 0.026 on average.

C.6 Motivating Experiments

Two comparisons show why the action allocation must go through model feedback. Figure 1(b) uses a selected Flights counterexample in which ground-truth (GT) repair at nearly matched weighted residual error reaches lower accuracy than an action allocation without additional GT repair, and uses Beers to examine the interaction between preparation and training-rule selection. The search surface, the initial GT controls and the counterexample selection are reported separately below.

Example C.1 (The Quality-State Search Space)**.** The Flights dataset (Mahdavi et al., 2019) records a data source, flight number, and scheduled and actual departure and arrival times. Four dependencies map flight to these time columns. The task predicts the three-valued arrival_delay_bucket with TabPFN-v2. The fixed experiment uses 252 training records at error rate $\rho = 0.4$. Varying repair coverage, deletion order, deletion fraction and positive weights produces 9,681 policies, including repair followed by downweighting. The search surface of Figure 1(b) uses the measured accuracy of all 9,681 policies. Each policy is post-trained once for three epochs with early stopping on 168 clean reference records and evaluated on the same 180 test records.

In this single-run evaluation, fixed mixed candidates #91 and #8449 achieve 56.67%, compared with 48.89% for the best of the four uniform single-action baselines, DeleteAll.

Example C.2 (Analysis of the Four Uniform Strategies)**.** In Example C.1, NoFix, DeleteAll, DownweightAll and RepairAll attain 41.11%, 48.89%, 46.67% and 45.00%, respectively. Here RepairAll applies validated candidates: 30 cell edits on 23 records reduce the error count from 955 to 927. DownweightAll preserves all values and assigns weight 0.25 to the 235 suspect records. DeleteAll retains 17 of the 252 training records at $(\kappa, \eta) = (0.027, 0.067)$, compared with $(0.541, 1.000)$ for NoFix. Deletion reduces both residual error and effective coverage.

Mixed #91 applies the same validated repairs and assigns weight 0.7857 to all 235 suspect records, retaining all 252 records. Mixed #8449 deletes 44 records by observed-label rarity, applies 27 validated edits to 21 surviving records and assigns weight 0.7857 to the 191 surviving suspect records. Both leave the 17 nonsuspect records at weight one. The weight is a rounded search-grid value, and both policies are fixed candidates. Their states are $(0.419, 0.996)$ and $(0.338, 0.821)$, respectively.

The surface measurements contain a two-by-two comparison of validated repair and a suspect-record weight on the same records. Relative to NoFix, RepairAll gains 3.89 points, the weight 0.786 alone gains 1.67, and their combination, Mixed #91, gains 15.56. The interaction of Equation (23), computed on accuracy gains, is $+10.00$ points, or 18 of the 180 test records. With the weights 0.25, 0.679 and 0.893 in place of 0.786, the interaction is -2.78, -0.56 and $+1.11$ points, or 5, 1 and 2 test records, respectively. At weight 0.893 the weight alone lowers accuracy by 3.33 points, while the combination raises it by 1.67. The repair gain and the weight gain do not add at weight 0.786, and the interaction at the other three weights lies within five test records of zero.

Ground-truth repair controls. For a mixed policy's target κ_t, GTRepair-κ samples $m_t = \text{round}(955 - nd\,\kappa_t)$ of the 955 erroneous training cells uniformly without replacement, including

label cells, and replaces their values with truth. It retains all $n = 252$ records with weight one and leaves other cells unchanged, so $\kappa_{\text{GT}} = (955 - m_t)/(nd)$, $\eta = 1$, and $|\kappa_{\text{GT}} - \kappa_t| \leq 1/(2nd)$, with $d = 7$. Identifying the error pool accesses all 252 training records and 1,764 true cell values, an acquisition cost distinct from the number of edits. These controls are separate from RepairAll's validated candidates and do not enter the search surface. Every policy is trained for three epochs with reference-based checkpoint selection on 168 records and evaluated on the same 180 test records. Their scores share an encoder that appends 19 feature categories from the clean training table, a separate diagnostic protocol from the surface scores.

The displayed counterexamples. Figure 1(b) displays the control for Mixed #8449: $m = 359$ GT edits give $\kappa_{\text{GT}} = 596/1764 = 0.33786848$ and $\eta_{\text{GT}} = 1$, against $\kappa = 0.33766037$ and $\eta = 0.82101066$ for Mixed #8449, which keeps 208 records. At nearly matched residual error and full coverage, GT repair reaches $79/180 = 43.89\%$ against $103/180 = 57.22\%$ for the mixed policy. The same comparison for Mixed #91 ($\kappa = 0.41872466$, $\eta = 0.996$) uses the minimum matching count of 217 edits and reaches $76/180 = 42.22\%$ against 57.22%. Even with access to the truth, repair at matched error and higher coverage can therefore lose to a mixed allocation.

Data and model decisions on Beers. The four configurations in Figure 1(b) use the experiment of Appendix C.3 and the last block of Table 1. Accuracy is 54.67% with the untouched data and the fixed training rule, 55.17% after preparation, 53.83% after training-rule selection on untouched data and 57.50% when the training rule is selected on prepared data. The resulting interaction is $+3.17$ percentage points. Thus, training-rule selection lowers accuracy on the untouched data but improves it on the prepared data. The four values report the reference-selected training runs of this comparison. The positive interaction shows that the value of a model-side decision depends on the prepared data.

C.7 Quality-state screening

Separating error reduction from retained coverage. The screening study compares candidate coordinates for Definition B.2, with residual error from Equation (3) and effective coverage from Equation (4), over ten ICL configurations, Adult, Breast Cancer, Flights, Hospitals and Tax at two error rates with 39,995 allocations in total. Residual error with effective coverage separates the four uniform strategies in eight of ten configurations, while class balance or label purity paired with coverage separates them in none.

C.8 Experimental Settings

Datasets. The main comparison uses six tabular datasets (Table 8). Beers, Flights, Hospitals and Tax are the cleaning benchmarks distributed with Raha and Baran (Mahdavi et al., 2019; Mahdavi & Abedjan, 2020). Adult (Becker & Kohavi, 1996) and Breast Cancer (Wolberg, 1990) come from the UCI repository. TEP (Downs & Vogel, 1993) is one additional dataset evaluated under six fault scenarios in Table 1. Labels with more than ten values are grouped into the nine most frequent classes and one remaining class.

Procedure. Each table is split 70/30 into fitting and test parts, stratified when class counts permit, and the fitting part supplies $N_{\text{ref}} = \min\{500, \lfloor 0.4 N_{\text{fit}} \rfloor\}$ clean reference records (Table 8). CARVEPrep screens its record-wise allocations within the cap, gives the first $B_f = 3$ candidates, NoFix and the Ref. likelihood and RHO-LOSS candidates three short-training runs each, and compares them by the adoption rule of Appendix D.3. Each adopted preparation then receives its own final training and clean-test evaluation.

Baselines. The uniform strategies are NoFix, RepairAll, DeleteAll, DownweightAll and RandomDelete. The cleaning comparisons are ActiveClean, a Picket-style reconstruction, Cleanlab, Raha+Baran, BoostClean, Jellyfish ED+DI and PriorAlign, and the selection comparisons are DSIR, EL2N, RHO-LOSS, Ref. likelihood, Data-OOB, Herding and GraNd. The seven selection rules share one encoded input and retain the top-ranked records up to the shared 0.3 record cap, and Ref. likelihood ranks by $\log p_{\text{ref}}(y_i \mid x_i)$ from a logistic model fitted on the reference set. The same score defines its candidates inside CARVEPrep. Each adaptation, its prompts and its hyperparameters are documented per method with the released code. For preparation, learned baselines receive the same

Table 8: Dataset sizes. Source is the full table size and Pool is the number of records used in the main comparison, sampled at random when the source exceeds 2,000 records. Brackets give the Adult and Tax pools used in the scale study. All six TEP scenarios use the listed pool. For Alpaca (Appendix C.5), Pool lists training records, with 200 reference and 500 test records held out separately.

Dataset	Source	Pool	Features	FDs	Classes	Kind	Downstream model
Adult (**Ad**)	45,222	2,000 (**1.25k–10k**)	14	1	2	Tabular	TabPFN-v2
Beers (**Be**)	2,410	2,000	3	0	10	Tabular	TabPFN-v2
Breast Cancer (**BC**)	699	699	9	0	2	Tabular	TabPFN-v2
Flights (**Fl**)	2,376	2,000	6	4	3	Tabular	TabPFN-v2
Hospitals (**Ho**)	1,000	1,000	18	13	5	Tabular	TabPFN-v2
Tax	200,000	2,000 (**1.25k–10k**)	11	3	2	Tabular	TabPFN-v2
Tennessee Eastman process (**TEP**)	960	960	52	0	2	Sequential	TabPFN-v2
Alpaca	5,000	3,500	–	–	–	Text	Qwen2.5-1.5B-Instruct (LoRA)

Table 9: Intervention costs averaged over the six tabular datasets. GT (Adult) is the number of clean records a method receives on Adult for preparation.

Metric	NoFix	Repair All	Delete All	Down-weightAll	Random Delete	Active Clean	Picket	Cleanlab
Acted	0.0	154.0	700.7	700.7	215.8	118.8	19.2	215.8
Changed cells	0.0	221.7	0.0	0.0	0.0	0.0	40.7	0.0
GT (Adult)	0	0	0	0	0	500	500	500

Metric	Prior Align	Raha+ Baran	Boost Clean	Jellyfish	DSIR	EL2N	RHO-LOSS	Ref. likelihood
Acted	568.2	702.2	690.7	697.7	215.7	215.7	215.7	215.7
Changed cells	2051.2	3459.3	2593.8	3552.2	0.0	0.0	0.0	0.0
GT (Adult)	500	500	500	500	500	500	500	500

Metric	Data-OOB	Herding	GraNd	Prior Only	Structural Only	CARVE Prep	Clean (oracle)
Acted	215.7	215.7	215.7	215.8	90.0	209.5	707.5
Changed cells	0.0	0.0	0.0	0.0	9.3	51.0	4005.8
GT (Adult)	500	500	500	500	500	500	900

clean reference records, the uniform strategies none, and the clean oracle the uncorrupted training split.

Metrics. A main-table cell is the test accuracy of the final model. Means weight datasets equally, and TEP scenarios are averaged separately. Repair error rate is the fraction of modified cells whose resulting values differ from the truth, averaged over the tasks with changed cells. It is 0.0817 for CARVEPrep and 0.5567 for Jellyfish. Table 9 reports the number of affected records, the number of changed cells and the number of clean records available to each method for preparation. A record with changed values, deletion or downweighting counts once as affected. Deletion changes no cell values, so the 70-fold difference measures rewrites. Affected-record means are 697.7 for Jellyfish and 209.5 for CARVEPrep, a 3.3-fold difference.

Injection. The mixture applies its rate separately to dependency violations, typos, missing values and label noise, so one record can contain several error types. The first three affect feature columns, and label noise affects the target column. Injecting into the clean releases controls the rate of each family independently and provides the cell-level ground truth behind the repair-error metric and the risk-control analysis. At $\rho = 0.4$, nearly every training record contains an error and a third to over half of all training cells are corrupted, so the 0.3 record cap reaches at most a third of the affected records.

Algorithm 2 Evidence Collection

Input: Table D, dependencies Σ, model M, optional reference D_{ref}, minimum candidate support and margin requirements
Output: Typed evidence, candidate values and validation outcomes
compute the reference-conditional record and cell priors, or use disjoint five-fold context scores without a reference
flag missing cells and form dependency groups on the observed table
for each $A \to B \in \Sigma$ and each eligible group **do**
    mark missing or inconsistent B cells, then validate the majority by group size, support and margin
    assign a candidate and directed tier where validation passes, and otherwise retain suspicion
end for
apply format and typo channels with their respective validation conditions
add numerical anomaly or temporal spike evidence
withdraw FD-violation candidates whose determinant cells are suspect
apply the label rule to the record prior
compute optional column-conditional priors when requested
return tiers, candidates, validation outcomes, s^{FM} and s^{DB}

Post-training. TabPFN-v2 (Hollmann et al., 2025) is post-trained with Adam at learning rate 10^{-5} for at most ten epochs and stops early on reference accuracy. Record weights enter the loss and context sampling as in Definition B.3.

D Deferred Method Details

Evidence, allocation and candidate evaluation. This section specifies the quantities used by Section 3, following the order in which the method uses them. Evidence channels produce the tiers and the validated values, value estimation and ranking turn those into one action per eligible record, candidate evaluation compares complete preparations, and the feedback and acceptance parts cover the iteration of Section 3.3.

D.1 Evidence channels

Evidence collection combines dependency witnesses, missing-value indicators, format and typo checks, and numerical anomaly scores (Algorithm 2). For $A \to B$, group records by their observed A values and count nonmissing B values. Let n_g be the count and c_1, c_2 the largest frequencies, taking $c_2 = 0$ if only one value occurs. The leading value becomes a repair candidate for missing or disagreeing cells when the group meets a minimum size, $c_1 > n_g/2$, $\frac{c_1}{n_g} \geq r_{\text{min}}$ and $\frac{c_1 - c_2}{n_g} \geq m_{\text{min}}$. After the format, typo and anomaly channels run, a dependency candidate is withdrawn when the record's determinant cell is suspect, and its violation evidence remains. A format candidate requires a content-preserving normalization to the dominant template of its column, a typo candidate requires a unique and much more frequent neighbor at edit distance one in a low-cardinality column, and the numerical channel flags robust outliers, contributing suspicion on a non-sequential table and an interpolation candidate checked for local temporal smoothness on a sequential one.

For classification with a labeled reference set, the record prior is

$$s_i^{\text{FM}} = \frac{\text{clip}\left(\log \frac{p(y_i)}{p(y_i \mid x_i)}, 0, \log C\right)}{\log C}, \qquad C = \max\{2, \text{number of reference classes}\}.$$

The probabilities are estimated from the reference data using the observed features. The label rule marks a record when $p(y_i \mid x_i) \leq \tau p(y_i)$, with $\tau = 1$ throughout, and Appendix E.2 distinguishes this likelihood-ratio criterion from the Bayes decision for actual label errors. Without a reference set, the prior is obtained by scoring each record with the other folds as context. Structural evidence s_{ij}^{DB} records dependency violation strength. A cell counts as suspect when it carries a witness tier or when its prior score or structural evidence reaches a fixed level. Proposition E.4 gives the correctness condition of the majority candidate and the recall limit of label-conditional scores.

D.2 Value estimation and ranking

Perturbation trials. Each trial of Section 3.2 splits the working table into a treatment pool and a scoring subset and injects additional missing values, dependency violations, typos and label redraws into the pool. Deletion, downweighting and repair then each receive several independent treated subsets drawn without replacement from the eligible suspect records, and an ICL evaluation on the scoring subset replaces a post-training run. The estimator is fitted on the contaminated working table and uses no ground truth.

Targets and regressors. With $\Delta_a(T)$ as defined in Section 3.2, every $i \in T$ receives the target $\frac{\Delta_a(T)}{|T|}$, a subset-average gain rather than an observed individual marginal effect. Each action has a gradient-boosted regressor on the features below, and retention has zero gain.

Record features. The features of a record describe evidence (tiers, validated candidates, unresolved violations, suspect cells and the record prior), statistical position (structural evidence and aggregated prior scores on suspect cells, column suspect rates, class share and value rarity) and the error channels that detected anomalies, and their definitions are shared across tables.

Ranking score. For d columns, let $T_i = \sum_j t_{ij}$ and $c_{\text{tier}} = 2d + 1$. Set $\widetilde{T}_i = 2c_{\text{tier}} + c_{\text{tier}} s_i^{\text{FM}}$ when $p(y_i \mid x_i) \le \tau p(y_i)$, and $\widetilde{T}_i = T_i$ otherwise. The default score is $s_i^{\text{rank}} = \mathcal{N}\Big(10\widetilde{T} + \mathcal{N}(s^{\text{FM}})\Big)_i$, where $\mathcal{N}$ is table-wise min-max normalization, with constant vectors mapped to zero. The factor 10 gives evidence the larger coefficient while the prior reorders nearby evidence totals (Definition B.5). The order $\prec_{\text{e}}$ sorts records by decreasing score with input order breaking ties, so label-rejected records precede feature-only evidence. The action rule of Section 3.2 applies within the ranked budget set. Every downweighted record, including a selected repair without a validated value, receives $w_i = \text{clip}(1 - \frac{3}{4} s_i^{\text{rank}}, w_{\min}, 1)$.

D.3 Candidate Evaluation

Candidates and scorer eligibility. Section 3.2 specifies the candidates and which of them are post-trained, and the short runs use a fixed number of epochs. A candidate that leaves too few positive-weight records or a single surviving class cannot be scored and is excluded from selection. On sequential tables the candidate set also contains interpolation and smoothing preparations (Appendix C.5).

Adoption rule. The bootstrap adoption rule uses mean-gain margin $m_{\text{gain}} = 0.01$ and a minimum paired-bootstrap win probability of 0.9 over the reference indices shared by the candidate and NoFix.

Execution of a candidate in the two settings. Fractional weights affect post-training loss and context sampling, as well as support draws during ICL screening (Definition B.3). For post-training, zero-weight rows are removed, the selected validated repairs are applied and the model minimizes the weighted loss of Definition B.3 with the context sampling stated there. In-context screening and the short training runs include each record in the support set with probability w_i and average the scores over three draws.

D.4 Feedback and Optional Proposals

Algorithm 3 states the loop of Section 3.3. The reported loop uses $R_{\text{loop}} = 3$ optimization iterations, and each selected candidate receives $R = 3$ short post-training runs (Appendix C.3, Figure 6(f)).

Algorithm 3 Alternating Preparation and Post-Training

Input: table D_0, dependencies Σ, reference D_{ref}, pretrained model M_0, optimization iterations R_{loop}
Output: Retained table and record weights, with the final post-trained model

```
initialize p_1, with selected preparation and reference scores ← none
for r = 1, ..., R_loop do
    prepare D_0 under p_r by collecting evidence, ranking and allocating, and for r > 1 add the retained table to the candidates
    select against D_ref over the candidates, giving D_r, which is the retained table when no candidate has a larger measured gain
    post-train M_0 on D_r with early stopping on D_ref, giving M_r
    compute losses under M_r with disjoint contexts, then update p_{r+1} by Appendix D.4
end for
train the final downstream model from M_0 on the retained preparation
return retained table and weights, with the newly trained downstream model
```

Record-loss feedback. For feedback, freeze the trained model parameters, split surviving records into five folds, and compute per-record cross-entropy using the other four folds as context, without the label base-rate adjustment of the initial prior in Section 3.2. Associate these losses with their original record indices and update $p_{r+1,i} = (1-\beta)p_{r,i} + \beta\mathcal{N}(\widetilde{\ell}_r)_i$ with $\beta = 0.5$ and the table-wise normalization $\mathcal{N}$ of Appendix D.2. Deleted records, which carry no loss, keep their previous priors. Records above the 0.8 quantile of the merged prior are assigned to the label-rejection category.

Verified value proposals. An optional model proposes values for unwitnessed suspect cells while retaining existing choices. Each complete candidate-value replacement rule is frozen before calibration, the repair error probability guarantee applies under the calibration sampling conditions of Lemma E.1, and proposal, verification and training costs count toward the budget. The reported experiments use the evidence channels without proposals.

D.5 Independent Calibration and Acceptance

The held-out variant uses development records for evidence collection, candidate construction, screening and early stopping, and fixes each prepared table, its weights and its fitted evaluation models before any acceptance record is read. The betting test of Proposition E.7 then compares the paired correctness scores of each candidate and NoFix, with its guarantee holding under Assumption B.1, and Appendix C.3 describes the split and the benchmark sampling conditions.

Before a replacement rule is applied to new records, Lemma E.1 calibrates its repair risk. It bounds the conditional error probability of a changed cell in a new record under a frozen complete replacement rule, using a verification sample with true values that is independent of development and acceptance. The reported repair error rates are the empirical fractions of Appendix C.8, measured against the known injected errors.

E Proofs

From representation to optimization and acceptance. We first establish that value configurations and weights cover the modeled preparation operators and characterize the budgeted search space. We then analyze what evidence can identify, how actions change attainable model performance and when an independent evaluation supports adoption. These results correspond to the representation, allocation and evaluation stages of the method.

E.1 Action Coverage and Search-Space Structure

Values and participation determine a preparation. We first characterize the four action primitives independently of how candidate values are obtained, then identify the outputs accessible to a fixed candidate menu. Two outputs are equivalent here when their weights agree and their values agree on every positive-weight record. Values at weight zero have no effect on either the loss or context of Definition B.3.

Proposition E.1 (Canonical representation of preparation outputs)**.** *Every output (D', w) of Definition B.1 has a unique representation, up to the above equivalence, by retention, deletion, repair, downweighting, or repair combined with downweighting. For fixed S and distinct configuration menus $\mathcal{Z}_i$, this output is represented by a policy of Definition 2.3 if and only if records outside S retain their original values and unit weights, and the output configuration of each positive-weight record in S belongs to $\mathcal{Z}_i$.*

Proof. At a record of weight zero, replace its stored values by the original values to obtain the canonical deletion. At positive weight there are two independent choices, whether the values change and whether the weight is one. Unchanged values with weight one give retention, changed values with weight one give repair, unchanged values with weight in $(0, 1)$ give downweighting, and changed values with weight in $(0, 1)$ give their combination. These cases are disjoint and exhaustive. They fix both the final values and weight, and hence the canonical representation. Membership in the menus and retention outside S are necessary by Definition 2.3. They are sufficient because each positive-weight output then has a unique configuration index, while each zero-weight output takes index zero. □

Thus imputation and label correction choose repair values, and selection or bounded importance weighting chooses participation, as in Example B.1. An external repair value enters the policy space when it is added to the relevant menu. This separates *coverage by the action primitives* from *coverage by the available candidates*.

The record budget couples the local choices. Let Π_b contain the canonical policies with at most b interventions and positive total weight, before imposing the repair-error constraint. At each record, every value configuration supplies a weight interval whose zero endpoint is identified with deletion. The global space is the product of these local choices restricted to at most b departures from retention.

Proposition E.2 (Budget layers and feasible reversal paths)**.** *The spaces are nested, $\Pi_0 \subseteq \Pi_1 \subseteq \cdots \subseteq \Pi_K$, with Π_0 containing only the unchanged table. If record i has $m_i = |\mathcal{Z}_i|$ configurations and a finite weight grid $\mathcal{W}_i \subseteq [0, 1]$ containing 0 and 1, put $q_i = m_i(|\mathcal{W}_i| - 1)$. Then*

$$|\Pi_b| = \sum_{j=0}^{b} [t^j] \prod_{i \in S} (1 + q_i t) - \mathbf{1}\{K = n = b\}, \tag{5}$$

where $[t^j]$ extracts a polynomial coefficient. After also imposing RepairError $\leq \alpha$*, join two feasible policies when they differ at one record. Every policy has graph distance exactly $|\mathcal{I}(\pi)| \leq b$ from the unchanged table, so this graph is connected and has diameter at most $2b$.*

Proof. Enlarging the intervention cap preserves every existing policy. Locally there is one deletion and $m_i(|\mathcal{W}_i| - 1)$ positive-weight outcomes, one of which is retention. Thus there are exactly q_i intervention choices. Selecting j intervened indices gives the coefficient of t^j in the product. The all-deletion policy is the only outcome with zero total weight, and it is counted exactly when S is the entire table and $b = n$.

For the path claim, reversing an intervention means restoring that record's original values and unit weight. This reduces its budget usage and cannot decrease total weight. For each repaired record i, let $u_i > 0$ count its changed cells and v_i count erroneous values among those changes. If any repairs remain, their aggregate error rate is $R = \sum_i v_i / \sum_i u_i$. Choose a record maximizing v_i/u_i, which is at least R, and reverse it. Removing a component whose ratio is at least the aggregate ratio cannot increase the aggregate, with the empty sum assigned rate zero. Repeating removes all repairs while preserving RepairError $\leq \alpha$. Reverse any remaining deletion or downweighting in any order, since neither changes the repair-error rate. Each step removes one intervened record, and every path to the unchanged table must change all $|\mathcal{I}(\pi)|$ such records. Its graph distance is therefore exactly $|\mathcal{I}(\pi)| \leq b$. Concatenating paths through the unchanged table gives the diameter bound. □

For equal $q_i = q$, Equation (5) becomes $\sum_{j=0}^{b} \binom{K}{j} q^j$ minus the possible all-deletion outcome. One repair candidate and one intermediate weight give five local outcomes, including the combined action, and $q = 4$. The reversal order matters under the repair-error constraint, since removing an accurate repair first can raise the error fraction of the remaining changes.

The budget also restricts the quality coordinates. Let e_i be the number of erroneous cells in original record i, and let T_b be the sum of the b largest e_i among $i \in S$, with $T_0 = 0$. Write $\kappa_0 = (nd)^{-1} \sum_i e_i$ for the original residual error.

Proposition E.3 (Budget envelope in the quality plane)**.** *Every policy in Π_b satisfies*

$$\kappa_0 - \frac{T_b}{nd} \le \kappa(D_\pi, w) \le \eta(w), \qquad \frac{\max\{n-b,1\}}{n} \le \eta(w) \le 1. \tag{6}$$

For $b < n$, deleting the b eligible records with largest e_i attains both lower bounds and has zero repair error. The lower coverage bound is also attainable when $b = n$ by retaining one record and deleting the rest.

Proof. Intervening on record i removes at most its original contribution $e_i/(nd)$ to residual error, so at most b interventions remove at most $T_b/(nd)$. Put $s = \sum_i w_i > 0$ and $v = \sum_i w_i^2 > 0$. Since $w_i \in [0,1]$, we have $v \le s$ and hence $\eta = s^2/(nv) \ge s/n \ge (n-b)/n$. Also $v \le s^2$ gives $\eta \ge 1/n$, while Cauchy–Schwarz gives $\eta \le 1$. A record has at most d erroneous cells, so $\kappa \le s/n \le \eta$. Deletion of the stated records proves attainment. □

These bounds describe how a record budget limits the joint change in residual error and effective coverage. The policy retains the identities and values of the affected records, which also determine downstream training. Downstream evaluation therefore uses the complete prepared values and weights.

E.2 Information Available to Detection

Proposition E.4 (Prediction information and structural recovery)**.** *Fix the prediction model and its reference information. Let $V \in \{0,1\}$ indicate an actual dependency violation, $\vartheta = \Pr(V=1) \in (0,1)$ and Z be the information a detector reads. Put $Q_v = \mathcal{L}(Z \mid V = v)$ and $\delta_Z = \mathrm{TV}(Q_1, Q_0)$. For selection fraction $\varrho \in [0,1]$, every measurable randomized rule $a(Z) \in [0,1]$ with $\mathbb{E}a(Z) \le \varrho$ obeys*

$$\mathrm{Recall}_Z(\varrho) \le \min\left\{1, \frac{\varrho}{\vartheta}, \varrho + (1-\vartheta)\delta_Z\right\}, \tag{7}$$

where recall is the supremum of $\mathbb{E}[a(Z) \mid V = 1]$ over these rules. There is an exact clean dependency $A \to B$ with marginal-resampling corruption for which $Z = (y, P_M(\cdot \mid x))$ has $Q_1 = Q_0$ even for a Bayes-optimal prediction model, while an intact group majority identifies every changed B cell.

More generally, fix before corruption a pool of N records and a complete dependency group of $m \ge m_{\mathrm{grp}}$ records, with $N \ge m$. The group's clean B values coincide, the determinant values in the pool remain unchanged throughout corruption, and the compared B values are nonmissing. The error indicators are either independent with probabilities at most ρ, or confined to K positions selected uniformly without replacement from the pool, with $\frac{K}{N} \le \rho$. For the support and margin thresholds of Appendix D.1, let

$$q_* = \min\left\{\frac{1}{2}, 1 - r_{\min}, \frac{1 - m_{\min}}{2}\right\}, \qquad 0 < \rho < q_*. \tag{8}$$

With probability at least

$$1 - \exp\{-m\,\mathrm{kl}(q_* \| \rho)\}, \qquad \mathrm{kl}(u\|v) = u \log\frac{u}{v} + (1-u)\log\frac{1-u}{1-v}, \tag{9}$$

the true value is the group's unique majority, passes both thresholds, and its minority test identifies exactly the changed cells. For sampling without replacement, a sharper recovery bound is

$$\sum_{\ell=0}^{\lceil m q_* \rceil - 1} \frac{\binom{K}{\ell}\binom{N-K}{m-\ell}}{\binom{N}{m}}, \tag{10}$$

with out-of-range binomial coefficients interpreted as zero.

Proof. Write $r_v = \mathbb{E}[a(Z) \mid V = v]$ and $q = \mathbb{E}a(Z)$. Total variation gives $r_1 - r_0 \le \delta_Z$, so $r_1 = q + (1-\vartheta)(r_1 - r_0) \le \varrho + (1-\vartheta)\delta_Z$. The mixture identity $q = \vartheta r_1 + (1-\vartheta)r_0$ also gives $r_1 \le \frac{\varrho}{\vartheta}$.

For the construction, take equally sized groups $A = 0, 1$ with $m \ge 3$ records, clean $B = A$ and independent fair binary labels. The model predicting $\frac{1}{2}$ everywhere is Bayes-optimal. Independently of the labels, resample one B per group from its binary marginal and sample a record uniformly. Then $\vartheta = \frac{1}{2m}$, while V is independent of both the label and the constant prediction, giving $Q_1 = Q_0$. Every group retains at least $m-1 > \frac{m}{2}$ true values, so comparison with its majority identifies exactly the changed cells. The same construction applies to a label law depending on A alone with the model $p(y \mid A)$.

For the probability bound, let N_e count erroneous values in the group. On $N_e < mq_*$ the true value has share $1 - \frac{N_e}{m} > \max\{\frac{1}{2}, r_{\min}\}$, and its margin over every other value is at least $1 - \frac{2N_e}{m} > m_{\min}$. This proves all three conclusions on that event. Under independent errors, their probability bound ρ gives, for $s > 0$,

$$\Pr(N_e \ge mq_*) \le e^{-smq_*}\mathbb{E}e^{sN_e} \le e^{-smq_*}(1 - \rho + \rho e^s)^m.$$

For sampling without replacement, let L count selected positions in the group, so $N_e \le L$. For $u = e^s - 1 \ge 0$, expansion over the m selection indicators gives

$$\mathbb{E}e^{sN_e} \le \mathbb{E}e^{sL} = \sum_{\ell=0}^{m} \binom{m}{\ell} u^\ell \frac{(K)_\ell}{(N)_\ell} \le \left(1 - \frac{K}{N} + \frac{K}{N}e^s\right)^m \le (1 - \rho + \rho e^s)^m,$$

where $(a)_\ell = a(a-1)\cdots(a-\ell+1)$ and $(a)_0 = 1$. The bound uses $\frac{(K)_\ell}{(N)_\ell} \le (\frac{K}{N})^\ell$, with zero numerator for $\ell > K$. Since L is hypergeometric, $\Pr(L < mq_*)$ gives Equation (10). Both sampling schemes therefore have the same exponential-moment bound. Taking $s = \log \frac{q_*(1-\rho)}{\rho(1-q_*)} > 0$ yields $\exp\{-m\,\mathrm{kl}(q_* \| \rho)\}$ and proves Equation (9). □

The dependency-only study in Appendix C.1 injects a fixed number of right-hand-side errors without replacement while preserving determinants, as in the second sampling model, and reports recall by median group size.

Interpretation of the label rule. Let $Y^\star \sim p(\cdot \mid x)$ and let an independent $C \sim \mathrm{Bernoulli}(\rho)$ trigger a redraw from the class marginal, independently of $Y^\star$ given x, and otherwise $Y = Y^\star$. For observed class y, put $q = p(y \mid x)$, $b = p(y) > 0$ and $d = (1-\rho)q + \rho b > 0$. Conditioning gives

$$\Pr(C = 1 \mid x, Y = y) = \frac{\rho b}{d}, \qquad \Pr(Y \ne Y^\star \mid x, Y = y) = \frac{\rho b(1-q)}{d}. \tag{11}$$

The factor $1 - q$ accounts for redraws that match the true label. Under equal costs for retaining a wrong label and deleting a correct one, the second posterior gives the Bayes deletion condition $(1 - \rho + 2\rho b)q \le \rho b$. CARVEPrep uses the reference estimate of $\frac{q}{b}$ as its ranking signal.

E.3 Exact Action Advantage in Weighted Post-Training

The previous subsection bounds what evidence can identify. It says nothing about what an identified error is worth to the model, which is the question this subsection answers for a family in which the answer is exact. We analyze how value changes and record weights alter the predictions attainable after fitting a linear model. Let $v_1, \ldots, v_m \in \mathbb{R}^m$ be linearly independent. For each j, training contains a fixed unit-weight anchor $(\sqrt{a_j}v_j, 0)$ with $a_j > 0$, and one eligible record initially equal to $(z_j^0 v_j, y_j^0)$ with weight one. Its finite set $\mathcal{V}_j$ of available value configurations (z, y) contains (z_j^0, y_j^0). Repair selects another configuration, deletion sets the weight to zero, and downweighting sets it between zero and one. At zero weight, values have their original representation. Repair combined with downweighting counts as one intervention. Anchors remain fixed, and at most $b \in \{0, \ldots, m\}$ eligible records may differ from their original configurations.

For policy π, the normalized weighted least-squares fit minimizes

$$L(\theta; \pi) = \frac{\sum_{j=1}^m a_j(\theta^\top v_j)^2 + \sum_{j=1}^m w_j(z_j\theta^\top v_j - y_j)^2}{m + \sum_j w_j}. \tag{12}$$

Write $q_j = \theta^\top v_j$. The basis is invertible, so these prediction coordinates vary independently, and the unique fitted values are

$$q_j(z, y, w) = \frac{wzy}{a_j + wz^2}, \qquad q_j^0 = \frac{z_j^0 y_j^0}{a_j + (z_j^0)^2}. \tag{13}$$

Let the task population satisfy $\Pr(X = v_j) = p_j > 0$, $\sum_j p_j = 1$, $\mathbb{E}[Y \mid X = v_j] = t_j$, and $\mathbb{E}Y^2 < \infty$. The population squared-loss excess risk is $R(q) = \sum_j p_j (q_j - t_j)^2$.

For a menu $\mathcal{M}_j$ of value-and-weight triples (z, y, w) containing $(z_j^0, y_j^0, 1)$, define its attainable predictions and largest local risk reduction by

$$\begin{aligned} S_j^{\mathcal{M}} &= \{q_j(z, y, w) : (z, y, w) \in \mathcal{M}_j\}, \\ h_j^{\mathcal{M}} &= p_j \left[(q_j^0 - t_j)^2 - \operatorname{dist}(t_j, S_j^{\mathcal{M}})^2\right] \geq 0. \end{aligned} \tag{14}$$

Assume each $S_j^{\mathcal{M}}$ is compact. Let $\operatorname{Top}_b(h)$ sum the b largest components, with $\operatorname{Top}_0(h) = 0$.

Theorem E.1 (Exact advantage under a record budget)**.** *For the learner in Equation (12), the minimum excess risk over configurations in $\mathcal{M}$ with at most b interventions is*

$$R_{\mathcal{M}}^\star(b) = R(q^0) - \operatorname{Top}_b(h^{\mathcal{M}}). \tag{15}$$

Consequently, for nested menus $\mathcal{M}_j^A \subseteq \mathcal{M}_j^B$ on every record,

$$R_A^\star(b) - R_B^\star(b) = \operatorname{Top}_b(h^B) - \operatorname{Top}_b(h^A) \geq 0. \tag{16}$$

For each value configuration, put $c_j(z, y) = \frac{zy}{a_j + z^2}$ and $C_j = \{0\} \cup \{c_j(z, y) : (z, y) \in \mathcal{V}_j\}$. The four action restrictions have attainable sets

$$\begin{aligned} S_j^{\text{RD}} &= \{0, q_j^0\}, \quad & S_j^{\text{no repair}} &= [\min(0, q_j^0), \max(0, q_j^0)], \\ S_j^{\text{no fractional}} &= C_j, & S_j^4 &= [\min C_j, \max C_j]. \end{aligned} \tag{17}$$

Here RD permits retention and deletion, no repair permits all weights in $[0, 1]$ with original values, no fractional permits all available value configurations with weights in $\{0, 1\}$, and the full set permits all available configurations with weights in $[0, 1]$.

Proof. The denominator of Equation (12) does not depend on θ. In the coordinates q, its numerator is a sum of strictly convex quadratics with coefficient $a_j + w_j z_j^2 > 0$, giving Equation (13). A policy changes at most b prediction coordinates, and the risk reduction in coordinate j is at most $h_j^{\mathcal{M}}$. Its total reduction is therefore at most $\operatorname{Top}_b(h^{\mathcal{M}})$. Conversely, select coordinates with the b largest reductions and take a nearest attainable prediction to t_j in each. Compactness ensures attainment. A zero reduction can be realized by the original configuration without spending budget. This policy attains the bound, proving Equation (15). Subtraction proves Equation (16).

For fixed (z, y), the derivative of $q_j(z, y, w)$ with respect to w is $\frac{a_j zy}{(a_j + wz^2)^2}$. Thus, as w varies from zero to one, the fitted prediction covers the interval between zero and $c_j(z, y)$. Every such interval contains zero, so their union is $[\min C_j, \max C_j]$. The other restrictions give the remaining sets in Equation (17). □

Along the nested comparison RD $\subseteq$ no fractional $\subseteq 4$, the exact advantage of Equation (16) separates into two nonnegative terms,

$$R_{\text{RD}}^\star(b) - R_4^\star(b) = \underbrace{\operatorname{Top}_b(h^{\text{no fractional}}) - \operatorname{Top}_b(h^{\text{RD}})}_{\text{adding value repair}} + \underbrace{\operatorname{Top}_b(h^4) - \operatorname{Top}_b(h^{\text{no fractional}})}_{\text{adding fractional weights}}. \tag{18}$$

E.3.1 Strict Separation with Continuous Weights

Proposition E.5 (Repair and weighting provide distinct gains)**.** *There is an eight-record instance of Equation (12) with intervention budget three for which the minimum population excess risks are*

$$R_4^\star(3) = 0, \qquad R_{\text{RD}}^\star(3) = \frac{5}{72}, \qquad R_{\text{no repair}}^\star(3) = \frac{1}{16}, \qquad R_{\text{no fractional}}^\star(3) = \frac{1}{144}. \tag{19}$$

Proof. For $j = 1, \dots, 4$, take the fixed anchor $(e_j, 0)$ and the eligible record $(z_j e_j, 1)$, with $a_j = 1$, $z^0 = (1, -1, -1, 1)$ and original weights one. The sole available repair changes z_2 from -1 to 1 and may be combined with downweighting. The true second and third eligible records are $(e_2, 1)$ and $(e_3, 0)$, respectively, and the other records are correct. Thus the available repair corrects a feature value. The anchors ensure a unique fit with

$$\widehat{\theta}_j = \frac{w_j z_j}{1 + w_j z_j^2}, \qquad R(\widehat{\theta}) = \frac{1}{4}\sum_{j=1}^{4}(\widehat{\theta}_j - t_j^0)^2, \qquad t^0 = \left(\frac{1}{2}, \frac{1}{2}, 0, \frac{1}{3}\right), \tag{20}$$

where $X = e_j$ uniformly and $Y \mid X = e_j \sim \text{Bernoulli}(t_j^0)$. The four eligible records can be retained, repaired, deleted and downweighted to $w_4 = \frac{1}{2}$, respectively. This changes three records and gives $\widehat{\theta} = t^0$.

With retention and deletion, the second coordinate is in $\{-\frac{1}{2}, 0\}$ and contributes at least $\frac{1}{16}$ to risk. The fourth is in $\{0, \frac{1}{2}\}$ and contributes at least $\frac{1}{144}$. Retaining records one and four and deleting two and three attains their sum $\frac{5}{72}$. Without repair, the second coordinate remains in $[-\frac{1}{2}, 0]$ even with continuous weights. Deleting records two and three and setting $w_4 = \frac{1}{2}$ attains risk $\frac{1}{16}$. Without fractional weights, the fourth coordinate is still in $\{0, \frac{1}{2}\}$. Repairing record two, deleting three and retaining the others attains risk $\frac{1}{144}$. All four optima obey the same budget of at most three interventions. □

For this instance, $R(q^0) = \frac{23}{72}$ and

$$h^4 = (0, \tfrac{1}{4}, \tfrac{1}{16}, \tfrac{1}{144}), \qquad h^{\text{RD}} = (0, \tfrac{3}{16}, \tfrac{1}{16}, 0).$$

Equation (18) assigns $\frac{1}{16}$ of the risk reduction to adding repair and $\frac{1}{144}$ to adding fractional weights at budget three. The latter gain uses a correctly observed record whose unit weight does not match its population contribution.

Corollary E.1 (Separation under perturbations)**.** *In Proposition E.5, keep the training labels, available configuration identities and the record budget fixed. Perturb each anchor feature vector, each original eligible feature vector and the available repaired vector by at most ζ in Euclidean norm. Let the target distribution remain uniform on $e_1, \dots, e_4$, with conditional means $t \in [0, 1]^4$ satisfying $\|t - t^0\|_\infty \le \eta$. If*

$$\zeta \le \frac{1}{1000}, \qquad \eta \le \frac{1}{100},$$

then each of the three restricted menus has optimal excess risk at least $\frac{1}{288}$ greater than that of the full menu. The feature perturbations may have arbitrary directions.

Proof. For any policy π, let $H_\pi \theta_\pi = b_\pi$ be the unnormalized normal equations at the original instance. We have $H_\pi = \text{diag}(1 + w_j) \succeq I$ and $\|\theta_\pi\|_2 \le 1$. For the perturbed matrices, all eight record weights are at most one, every original feature vector has norm one, and only the four eligible labels are nonzero. Hence, uniformly over all policies, including continuous weights,

$$\|\widetilde{H}_\pi - H_\pi\|_{\text{op}} \le 16\zeta + 8\zeta^2 =: A_\zeta, \qquad \|\widetilde{b}_\pi - b_\pi\|_2 \le 4\zeta.$$

Since $A_\zeta < 1$, the perturbed fits are unique. Subtracting their normal equations gives

$$\|\widetilde{\theta}_\pi - \theta_\pi\|_2 \le \frac{4\zeta + A_\zeta}{1 - A_\zeta} = \frac{20\zeta + 8\zeta^2}{1 - 16\zeta - 8\zeta^2} =: \kappa_\zeta. \tag{21}$$

Put $e = \frac{\kappa_\zeta}{2} + \eta$. At the stated radii, $e \le \frac{248049}{12299900} < \frac{1}{48}$, which is smaller than $\sqrt{r}$ for each restricted menu. The original full-menu optimizer has perturbed risk at most e^2. For a restricted menu whose original minimum is r, every perturbed prediction has distance at least $\sqrt{r} - e$ from the new target in the norm $\frac{1}{2}\|\cdot\|_2$. Consequently,

$$\widetilde{R}^\star_{\text{restricted}} - \widetilde{R}^\star_4 \ge r - 2\sqrt{r}\, e. \tag{22}$$

The function $r - 2\sqrt{r}\, e$ is increasing for $r \ge \frac{1}{144}$. Each r in Equation (19) for a restricted menu is at least $\frac{1}{144}$, so the right-hand side is at least $\frac{1}{144} - 2 \cdot \frac{1}{12} \cdot \frac{1}{48} = \frac{1}{288}$. □

E.3.2 Interaction of Record Actions

Theorem E.1 places one eligible record on each basis direction. Each record then determines its own prediction, the record cap is the only coupling, and the risk reductions of separate records add. When several eligible records lie on one direction, the fitted prediction depends on their joint assignment. For a policy π let q_π denote its fitted prediction and $\Delta(\pi) = R(q^0) - R(q_\pi)$ its risk reduction. An edit is a policy that changes the configurations or weights of a set of eligible records and retains all other records. For two edits σ and τ that change disjoint record sets, or the values and the weights of the same records, let $\sigma\tau$ apply both. Their interaction is

$$I(\sigma, \tau) = \Delta(\sigma\tau) - \Delta(\sigma) - \Delta(\tau). \tag{23}$$

Proposition E.6 (Record actions interact)**.** *Extend the fit of Equation (12) to one direction with a unit anchor and six eligible records $(z_i v_1, y_i)$, the second sum being taken over these records. There is an instance in which deleting three mislabeled records, the edit σ, raises the excess risk from $\frac{1}{196}$ to $\frac{1}{4}$, repairing the feature value of another record, the edit τ, raises it from $\frac{1}{196}$ to $\frac{9}{196}$, and the two edits together attain zero excess risk. In this instance $\Delta(\sigma) < 0$, $\Delta(\tau) < 0$, $\Delta(\sigma\tau) > 0$ and $I(\sigma, \tau) = \frac{57}{196}$.*

Proof. Take $m = 1$, $v_1 = 1$ and $a_1 = 1$, and let $Y \mid X = v_1 \sim$ Bernoulli$(\frac{1}{2})$, so that $t_1 = \frac{1}{2}$. The eligible records (z_i, y_i) are $(1, 1)$ and $(1, 0)$, three records $(1, 1)$ whose true label is 0, and one record $(-1, 1)$ whose true feature value is 1 and whose available repair is $(1, 1)$. The objective is $L(q; \pi) = \frac{q^2 + \sum_i w_i (z_i q - y_i)^2}{1 + \sum_i w_i}$. Its denominator does not depend on q, and the unique minimizer is

$$q = \frac{\sum_i w_i z_i y_i}{1 + \sum_i w_i z_i^2}. \tag{24}$$

The original table gives $q^0 = \frac{3}{7}$. Deleting the three mislabeled records, the edit σ, gives $q = 0$. Repairing the record $(-1, 1)$, the edit τ, gives $q = \frac{5}{7}$, and both edits give $q = \frac{1}{2}$. The excess risks $(q - \frac{1}{2})^2$ are $\frac{1}{196}$, $\frac{1}{4}$, $\frac{9}{196}$ and 0, so $\Delta(\sigma) = -\frac{12}{49}$, $\Delta(\tau) = -\frac{2}{49}$ and $\Delta(\sigma\tau) = \frac{1}{196}$. □

The mislabeled records bias the prediction upward and the sign error biases it downward. The two errors partly cancel in the original table, and correcting one of them alone removes this cancellation. The repair changes the risk reduction by $\Delta(\tau) = -\frac{2}{49}$ when applied to the original table and by $\Delta(\sigma\tau) - \Delta(\sigma) = \frac{1}{4}$ when applied after the deletions, and the two values differ by $I(\sigma, \tau)$. An edit has the same marginal gain in either order exactly when the interaction vanishes, as it does for edits of separate records in Theorem E.1. The difference of optimal risks between nested menus in Equation (16) is nonnegative with or without additivity, because the larger menu contains every policy of the smaller one. The experiments report measured counterparts of this comparison, nested candidate sets under one adoption rule (Table 3) and the best evaluated candidates of nested action sets (Figure 6(b)). In the candidate-set comparison, adoption replaces optimal selection, so nestedness does not determine the sign of the measured difference. Example C.2 reports a measured two-by-two comparison on Flights.

Implications for candidate evaluation. Section 3.2 states what these three results imply for the method. The exact budgeted advantage and the strict separation justify the four-action candidate set, and the non-additive instance is why record-wise estimates decide actions while the measured accuracy of complete preparations decides adoption. The comparison of nested action sets requires no additivity, because the larger set contains every policy of the smaller one.

E.4 Statistical Validity of Independent Acceptance

The two preceding subsections describe what can be detected and what an action set is worth. Neither controls the risk that a candidate measured as better on a finite reference set is worse in population, which is what the following result bounds. It applies the standard betting construction and Ville's inequality to candidate comparison (Waudby-Smith & Ramdas, 2024; Howard et al., 2021; Ville, 1939).

Proposition E.7 (Acceptance of frozen scorers)**.** *Condition on all information fixed before acceptance, including development and any optional calibration, and fix $K_c \geq 1$ candidate score functions $g_1, \ldots, g_{K_c}$ and a reference score g_0, each taking values in $[0,1]$. The count K_c and all fitted models are fixed before accessing an independent i.i.d. acceptance sequence $X_1, X_2, \ldots$ from the task population, independently of all fixed information (Assumption B.1). Set $d_{k,t} = g_k(X_t) - g_0(X_t)$ and $\mu_k = \mathbb{E}[d_{k,t}]$. For $0 \leq \epsilon < 1$ and $\delta_2 \in (0,1)$, define*

$$E_{k,0} = 1, \qquad E_{k,t} = \prod_{s=1}^{t} \{1 + \nu_{k,s}(d_{k,s} + \epsilon)\}, \qquad 0 \leq \nu_{k,s} \leq \frac{1}{2(1-\epsilon)}. \tag{25}$$

Each bet $\nu_{k,s}$ may use the information fixed before acceptance and all candidates' scores on records $1, \ldots, s-1$, but not record s or later records. Adopt a candidate only after its process reaches $\frac{K_c}{\delta_2}$, and retain the reference if none qualifies. Then

$$\Pr\left(\exists k, t : \ \mu_k < -\epsilon, \ E_{k,t} \geq \frac{K_c}{\delta_2}\right) \leq \delta_2. \tag{26}$$

Thus any selection among qualifying candidates, including selection at a data-dependent stopping time, has population score at least that of the frozen reference minus ϵ, except on an event of probability at most δ_2. With no candidates, retain the reference.

Proof. Condition on the information fixed before acceptance, and let $\mathcal{F}_{t-1}$ be the σ-field generated by that information and records $X_1, \ldots, X_{t-1}$. For a candidate with $\mu_k \leq -\epsilon$, independence and predictability give

$$\mathbb{E}[1 + \nu_{k,t}(d_{k,t} + \epsilon) \mid \mathcal{F}_{t-1}] = 1 + \nu_{k,t}(\mu_k + \epsilon) \leq 1.$$

Since $d_{k,t} + \epsilon \geq -(1-\epsilon)$, every factor in Equation (25) is at least $\frac{1}{2}$ and at most $1 + \frac{1+\epsilon}{2(1-\epsilon)}$. Hence $E_{k,t}$ is a nonnegative supermartingale starting at one. Ville's inequality bounds its crossing probability by δ_2/K_c. A union bound over the fixed candidates and removal of the conditioning prove Equation (26). Candidate processes may share the same acceptance records. □

E.4.1 Transfer to the final retrained classifier

Proposition E.7 controls the scores of the evaluation classifiers that were frozen before acceptance, and the classifier trained on the adopted preparation is not one of them. The result below transfers the guarantee to that classifier through a confidence bound on how often the two disagree on unlabeled features.

Corollary E.2 (Final-fit transfer by prediction disagreement)**.** *Assume the conditions of Proposition E.7, with candidate scores*

$$g_k(x,y) = \frac{1}{R}\sum_{r=1}^{R} \mathbf{1}\{f_{k,r}(x) = y\}, \qquad R \geq 1,$$

where all evaluation classifiers are fixed before acceptance. Let A be the event that a candidate is adopted, and on A let J be the selected candidate and f'_J the classifier retrained on its preparation. Fix the final classifier before drawing a fixed number $m \geq 1$ of independent unlabeled features $U_1, \ldots, U_m$ from the feature marginal P_X of the task distribution P, independently of all data and randomness used for selection and training. For fixed $\delta_3 \in (0,1)$, define

$$\begin{aligned} D_J(x) &= \frac{1}{R}\sum_{r=1}^{R} \mathbf{1}\{f'_J(x) \neq f_{J,r}(x)\}, \qquad \widehat{q} = \frac{1}{m}\sum_{i=1}^{m} D_J(U_i), \\ q_U &= \sup\left\{ q \in [\widehat{q}, 1] : m\,\mathrm{kl}(\widehat{q} \| q) \leq \log\frac{1}{\delta_3} \right\}. \end{aligned} \tag{27}$$

Here kl *is the binary relative entropy in Equation (9), with its usual limiting values at zero and one. In particular, $q_U = 1$ when $\widehat{q} = 1$. Writing $P(g) = \mathbb{E}_{(X,Y)\sim P} g(X,Y)$ and $\mathrm{Acc}_P(f'_J) = \Pr_{(X,Y)\sim P}(f'_J(X) = Y)$ for the realized final classifier,*

$$\Pr(A \cap \{\mathrm{Acc}_P(f'_J) < P(g_0) - \epsilon - q_U\}) \leq \delta_2 + \delta_3. \tag{28}$$

Proof. Condition on the complete selection and training history $\mathcal{H}$ for which A occurs. The functions D_J and f'_J are fixed. Write $q = \mathbb{E}_{P_X} D_J(X)$. For $\lambda < 0$, convexity on $[0,1]$ gives $\mathbb{E}[e^{\lambda D_J(U_i)} \mid \mathcal{H}] \le 1 - q + qe^{\lambda}$. Applying the exponential Markov inequality and optimizing over λ yields

$$\Pr(\widehat{q} \le x \mid \mathcal{H}) \le \exp\{-m\,\mathrm{kl}(x\|q)\}, \qquad 0 \le x < q.$$

Inverting this bound gives $\Pr(q > q_U \mid \mathcal{H}) \le \delta_3$. The cases $q = 0$ and $q = 1$ hold directly. For every (x, y),

$$\mathbf{1}\{f'_J(x) = y\} \ge g_J(x,y) - D_J(x).$$

Taking expectations yields $\mathrm{Acc}_P(f'_J) \ge P(g_J) - q$. Proposition E.7 bounds the probability of adopting a candidate with $P(g_J) < P(g_0) - \epsilon$ by δ_2. Removing the conditioning and taking a union bound proves Equation (28). □

The bound q_U is computed by a one-dimensional monotone search. Since $\mathrm{kl}(x\|q) \ge 2(q-x)^2$ for $q \ge x$, it satisfies $q_U \le \min\{1, \widehat{q} + \sqrt{\frac{1}{2m}\log\frac{1}{\delta_3}}\}$. When the final classifier and every evaluation classifier agree on all sampled features, $\widehat{q} = 0$ and $q_U = 1 - \delta_3^{\frac{1}{m}}$, which decreases at order $\frac{1}{m}\log\frac{1}{\delta_3}$.

E.4.2 Calibration of replacement rules

Fixed-sequence testing (Angelopoulos et al., 2021) on independently verified cell values controls repair error. The lemma specifies the sampling unit when a record contains several changed cells.

Lemma E.1 (Repair-risk calibration from independent records)**.** *Freeze $L \ge 1$ complete replacement rules, their order and all allocation context before drawing a fixed sample of $h \ge 1$ independent, identically distributed verification records $Z_1, \dots, Z_h$ from the deployment population, independently of that frozen information. Each record has $d \ge 1$ cells, and each rule acts on a record using only that record and the frozen context. Let $M_j(Z)$ count cells actually changed by rule j, and let $C_j(Z)$ count those changes whose resulting values differ from the truth. Truth is used only to verify changes, not to choose them. Set*

$$A_j = \frac{M_j}{d}, \qquad B_j = \frac{C_j}{d}, \qquad R_j^{\mathrm{cell}} = \frac{\mathbb{E}B_j}{\mathbb{E}A_j},$$

with $R_j^{\mathrm{cell}} = 0$ when $\mathbb{E}A_j = 0$. For fixed $\alpha, \delta \in (0,1)$, let $\widehat{A}_j = h^{-1}\sum_{t=1}^{h} A_j(Z_t)$ and define $\widehat{B}_j$ analogously. Inspect the rules in their fixed order, continue while

$$\widehat{B}_j - \alpha\widehat{A}_j + \sqrt{\frac{1}{2h}\log\frac{1}{\delta}} \le 0, \tag{29}$$

and stop at the first failure. If $\mathcal{J}_{\mathrm{pass}}$ contains the rules passed before stopping, then

$$\Pr\big(\exists j \in \mathcal{J}_{\mathrm{pass}} : R_j^{\mathrm{cell}} > \alpha\big) \le \delta.$$

Any passing rule may be returned, and if none passes, make no changes. Rules with $M_j \equiv 0$ have $R_j^{\mathrm{cell}} = 0$ and are admitted directly, and the fixed-sequence test applies to the remaining rules in their inherited order.

Proof. Condition on the frozen rules and context. Since $0 \le B_j \le A_j \le 1$, the variables $W_{j,t} = B_j(Z_t) - \alpha A_j(Z_t)$ are independent across records and lie in $[-\alpha, 1-\alpha]$, an interval of width one. For any rule with $R_j^{\mathrm{cell}} > \alpha$,

$$\mu_j = \mathbb{E}W_{j,t} = \mathbb{E}A_j\,(R_j^{\mathrm{cell}} - \alpha) > 0.$$

Writing $c_h = \sqrt{\frac{1}{2h}\log\frac{1}{\delta}}$, Hoeffding's inequality (Hoeffding, 1963) gives

$$\Pr(\widehat{B}_j - \alpha\widehat{A}_j + c_h \le 0) \le \Pr\left(h^{-1}\sum_t W_{j,t} - \mu_j \le -c_h\right) \le e^{-2hc_h^2} = \delta.$$

If any bad rule passes, the first rule in the fixed order with $R_j^{\mathrm{cell}} > \alpha$ must pass. Applying the preceding bound to this fixed first bad rule proves the claim. Removing the conditioning preserves the probability bound. Finally, choose a column uniformly from $\{1, \dots, d\}$, independently of a new record. Its probabilities of being changed and being incorrectly changed are $\mathbb{E}A_j$ and $\mathbb{E}B_j$, respectively. Thus R_j^{cell} is exactly its error probability conditional on an actual change. □

If at most $K_{\text{cal}} \geq 1$ replacement-rule families are fixed before calibration, apply Lemma E.1 to each at level $\frac{\delta_1}{K_{\text{cal}}}$, including families that are rejected. Restrict subsequent acceptance to passing rules or rules with no value changes. With calibration independent of development and acceptance, a union bound with Proposition E.7 gives simultaneous repair risk at most α and frozen-score loss at most ϵ with probability at least $1 - \delta_1 - \delta_2$. Rules that recompute rankings, quotas or dependency majorities from the verification sample fall outside the lemma.