RESIDUAL ANALYTICAL LEARNING FOR AUDITABLE REGRESSION CORRECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

A fitted regressor may leave systematic errors whose correction depends on a much smaller subset of variables than the original prediction task. This observation motivates a different way to improve an existing regressor by preserving the fitted model and learning an explicit correction only for the structure it leaves behind. We develop Residual Analytical Learning, a framework that learns from out-of-fold residuals, identifies low-dimensional residual structure, and fits analytical correction laws within supported error regimes. The resulting rule is not a post-hoc explanation of another model. It is the computation actually added to the base prediction, providing an explicit instance-level account of how and why that prediction was changed. Across eight heterogeneous regression benchmarks, the proposed method improves the frozen linear base in all 40 outer folds, with dataset-level MAE reductions ranging from 9.6% to 97.0% and a median reduction of 26.5%. Under a correction search capped at four variables, seven of eight benchmarks select a median residual dimension of at most three, including a stable two-variable residual core on a 32-dimensional benchmark. This analytical structure comes with an accuracy–auditability trade-off. Flexible residual learners remain more accurate on most datasets, while the proposed method recovers a median 55.0% of the improvement achieved by the best tested residual corrector while retaining the exact computation responsible for each correction. We further find that learning systematic residual structure and predicting the error that remains after correction are distinct problems.

1 INTRODUCTION

In tabular regression, predictive accuracy and auditability often pull in different directions. Simple models such as linear regression expose their parameters and computations directly, which is valuable when predictions must be inspected or justified (Rudin, 2019), whereas flexible models, particularly tree-based ensembles, often achieve stronger predictive performance (Grinsztajn et al., 2022). This creates a practical problem when a model is already trained, validated, and embedded in a workflow: replacing it may improve accuracy but remove the ability to trace how individual predictions were produced.

Existing work largely follows two directions. Post-hoc methods such as LIME (Ribeiro et al., 2016) and SHAP (Lundberg & Lee, 2017) explain fitted models without changing them, but the explanation remains separate from the computation that generated the prediction and may itself be unstable or manipulated (Baniecki & Biecek, 2024). Alternatively, intrinsically interpretable models such as generalized additive models (Hastie & Tibshirani, 1990) and neural additive models (Agarwal et al., 2021; Molnar, 2022) provide transparency by design, but require replacing the original predictor.

Related calibration and smoothing methods show that fitted model outputs can be adjusted without replacing the underlying predictor. Isotonic regression (Zadrozny & Elkan, 2002) and Platt scaling (Platt, 1999) calibrate model outputs, while LOWESS (Cleveland, 1979) can capture smooth systematic error and feature-dependent tree correction (Molnar, 2022) can adapt corrections across the input space. More generally, boosting methods learn successive residual-like updates and can do so highly flexibly (Friedman, 2001; Chen & Guestrin, 2016; Ke et al., 2017). Yet structured residual variation need not require temporal drift or online adaptation. Even in a fixed regression problem, errors can

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.

vary systematically across the input space. This suggests an offline alternative: learn recurring error structure from training data itself, while leaving the deployed predictor unchanged.

Building on this view, we develop Residual Analytical Learning (RAL), an explicit correction layer for a frozen regressor. Given a fitted base model f_0, RAL constructs out-of-fold residuals, screens variables and pairwise interactions for residual structure, searches for a low-dimensional residual subspace, and partitions it into supported regions. Within each region, the correction is selected from constant, linear, hinge, monotone piecewise-linear, and local-affine analytical laws. Cross-fitted evaluation selects the law and shrinkage level, while an order-independent subset lattice allows backoff to a lower-dimensional supported rule when a more specific region is sparse.

The final prediction is the frozen base prediction plus the analytical correction $\Delta(x)$ selected for the instance. Crucially, this is not a surrogate explanation of another correction model: it is the computation actually added to the base prediction. The system can therefore expose the variables used, residual region, analytical law, parameters, and numerical adjustment while keeping the original prediction visible.

RAL also separates two quantities often conflated as "confidence." *Correction utility* asks whether an analytical adjustment is supported relative to retaining the base prediction; *final prediction risk* estimates the magnitude of the error expected to remain afterward. The former controls whether a correction is applied, while the latter supports risk stratification, selective prediction, and locally scaled conformal intervals. Selective prediction uses estimated risk to retain or defer predictions (Geifman & El-Yaniv, 2019), while split conformal prediction provides finite-sample marginal coverage under exchangeability (Lei et al., 2018); conformalized quantile regression further illustrates how local error information can improve interval adaptivity (Romano et al., 2019). Here, conformal inference is used not as a new validity result, but to test whether post-correction risk contains useful information about the error that remains.

We evaluate one frozen configuration on eight heterogeneous regression benchmarks using five outer folds and separate proper-training, correction/risk-calibration, and conformal-calibration subsets. All residual correctors use the same frozen linear base specification and the same five-fold OOF residual-construction procedure. Our contributions are:

- We introduce RAL, which preserves a frozen regressor and learns explicit analytical corrections from out-of-fold residual structure.
- We develop a low-dimensional correction mechanism combining residual screening, supported regions, analytical law selection, cross-fitted shrinkage, and subset-lattice backoff.
- We make the deployed correction itself auditable and separate evidence for applying a correction from post-correction absolute-error risk.

2 Residual Analytical Learning and Experimental Protocol

Residual Analytical Learning (RAL) preserves a fitted regressor and learns an explicit correction for systematic error left by that model. Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$, with $x_i \in \mathbb{R}^p$ and target y_i, and let $\hat{y}_0(x) = f_0(x)$ be the frozen base prediction. RAL produces

$$\hat{y}(x) = \hat{y}_0(x) + \Delta(x), \tag{1}$$

where $\Delta(x)$ is generated by an analytical law whose inputs, parameters, and numerical contribution are recoverable for the individual prediction.

2.1 Residual representation and analytical correction

We use five outer folds. Within each outer-training fold, data are split into 80% proper training, 10% correction/risk calibration, and 10% conformal calibration; the outer-test fold is used only for evaluation. Five-fold cross-fitting on proper training gives OOF base predictions $\tilde{y}_{0,i}$ and residual targets

$$r_i = y_i - \tilde{y}_{0,i}. \tag{2}$$

Thus residual structure is learned from errors produced by models that did not train on the corresponding observations. The base specification is chosen by a fixed linear/Ridge rule and refitted on all proper-training data; ordinary linear regression is selected in every reported outer fold.

RAL searches for variables that organize r_i, rather than for a second representation of y. Stability-fold mutual information and a pairwise interaction screen form a candidate pool of at most 12 variables. Candidate subsets $S = \{s_1, \ldots, s_d\}$ use $d \in \{1, 2, 3, 4\}$ and empirical quantile bin counts $B \in \{3, 4, 5, 6, 7, 8, 10\}$. Their Cartesian bin combinations define residual regions, with poorly supported configurations discarded. Remaining candidates are ranked by OOF regional correction using

$$J_{\text{coarse}} = \text{MAE}_{\text{coarse}} + \text{MAE}_0 \left[\lambda_d d + \lambda_R \frac{\log(1+K)}{\log(1+n_s)} \right], \tag{3}$$

where $\text{MAE}_{\text{coarse}}$ and MAE_0 are corrected and uncorrected OOF MAE, K is the number of observed fine regions, and n_s the search-sample size; $\lambda_d = 0.001$ and $\lambda_R = 0.002$. The strongest configurations are reranked using analytical correction, yielding the selected S and B. Complete search settings are reported in Appendix A.

RAL then builds an order-independent subset lattice over S, allowing a prediction to back off from a sparse full-dimensional region to a supported lower-dimensional rule. At each supported node $\mathcal{G}_{A,k}$, $A \subseteq S$, it considers constant, linear, hinge, monotone piecewise, and local-affine laws. The local-affine family is

$$g_{A,k}(x) = \beta_0 + \beta_y \frac{\hat{y}_0(x) - \mu_y}{\sigma_y} + \sum_{j \in A} \beta_j \frac{x_j - \mu_j}{\sigma_j}, \tag{4}$$

where μ_y, σ_y and μ_j, σ_j are fitting statistics for the base prediction and selected variables. A ridge penalty of 10^{-4} stabilizes the fit.

For each law family m, five-fold cross-fitting produces OOF corrections $\tilde{g}_{i,m}$. The family and shrinkage $\alpha \in \mathcal{A} = \{0, 0.05, \ldots, 1\}$ minimize

$$(\hat{m}, \hat{\alpha}) = \arg \min_{m \in \mathcal{F},\, \alpha \in \mathcal{A}} \frac{1}{|\mathcal{I}_{A,k}|} \sum_{i \in \mathcal{I}_{A,k}} |r_i - \alpha \tilde{g}_{i,m}|, \tag{5}$$

where $\mathcal{F}$ is the analytical-law family and $\mathcal{I}_{A,k}$ contains proper-training observations in the region. Candidates with $\hat{\alpha} = 0$ or no OOF improvement are discarded. A separate constant-residual policy provides a conservative fallback. Inputs are clipped to fitted ranges and corrections to proper-training residual bounds; resulting extrapolation penalties affect held-out evidence, not the analytical law itself.

2.2 Held-out selection, risk, and conformal inference

The correction/risk-calibration subset determines which frozen candidate is deployed. For candidate prediction $\hat{y}_i$ we record

$$\begin{aligned} H_i &= \mathbf{1}[|y_i - \hat{y}_i| < |y_i - \hat{y}_{0,i}|], \\ G_i &= |y_i - \hat{y}_{0,i}| - |y_i - \hat{y}_i|, \\ E_i &= |y_i - \hat{y}_i|, \end{aligned} \tag{6}$$

where H_i indicates improvement over the base, G_i is absolute-error gain, and E_i is corrected absolute error. Supported states provide one-sided 95% lower bounds for correction success and mean gain and an upper bound for mean corrected error, with hierarchical pooling when necessary. An analytical rule is eligible when its adjusted success lower bound exceeds 0.50 and its gain lower bound is positive; the eligible rule with the smallest conservative corrected-error upper bound is selected. If none passes, a separately trained constant-residual fallback is tested at a 0.55 success threshold; otherwise the base prediction is retained.

The deployed prediction therefore has the explicit record

$$C(x) = \{\hat{y}_0(x), A_x, \mathcal{G}_x, g_x, \Delta(x), \hat{y}(x), U(x), \mathcal{R}(x), I_\alpha(x)\}\,, \tag{7}$$

where A_x, $\mathcal{G}_x$, g_x, and $U(x)$ denote the selected residual subset, region, analytical law, and held-out correction evidence. Importantly, g_x is the rule that generates the numerical correction, not a surrogate explanation of another corrector.

Post-correction risk is estimated only after this policy is frozen. Let $R_{\text{state}}(x)$ be calibrated mean absolute error for the most specific supported policy state and $M_{\text{OOF}}(x)$ its proper-training OOF error. RAL predicts

$$\mathcal{R}(x) = wR_{\text{state}}(x) + (1-w)M_{\text{OOF}}(x), \tag{8}$$

where $w \in \{0, 0.1, \ldots, 1\}$ is selected on the correction/risk-calibration subset. Unsupported states use hierarchical pooling, and calibration quartiles define Very Low, Low, Moderate, and High predicted-risk groups.

The untouched conformal subset is used only after correction and risk policies are fixed. Its normalized conformity scores are

$$s_i = \frac{|y_i - \hat{y}_i|}{\max\{\mathcal{R}(x_i), \varepsilon\}}, \tag{9}$$

and, for finite-sample split-conformal quantile $q_{1-\alpha}$,

$$I_\alpha(x) = [\hat{y}(x) - q_{1-\alpha}\mathcal{R}(x),\ \hat{y}(x) + q_{1-\alpha}\mathcal{R}(x)]\,. \tag{10}$$

Risk-normalized intervals are compared with ordinary split conformal based on unnormalized absolute errors. Under exchangeability, the usual marginal split-conformal guarantee applies; no conditional-coverage claim is made (Lei et al., 2018). Full training and inference algorithms appear in Appendix B.

2.3 Evaluation

We evaluate one frozen configuration on eight regression benchmarks spanning synthetic, physical-system, housing, computer-activity, robot-dynamics, and protein-structure tasks. Dataset size ranges from 1,503 to 45,730 observations and dimensionality from four to 32 predictors.

Table 1: Regression benchmarks and selected residual structure. Residual dimension is the median across five outer folds.

Dataset	n	p	Median d	Repeated residual core
CCPP	9,568	4	3	V, AP (5/5)
Friedman2	10,000	4	2	x_2, x_3 (5/5)
Airfoil	1,503	5	3	frequency, chord (5/5)
California	20,640	8	2	longitude, latitude (5/5)
CPU activity	8,192	21	3	freeswap, runqsz (5/5)
Kin8nm	8,192	8	4	$\theta_4, \theta_5, \theta_6, \theta_7$ (5/5)
Protein	45,730	9	3	F3, F8 (5/5)
Pumadyn32nh	8,192	32	2	tau4, theta5 (5/5)

CCPP, Airfoil, and Protein are UCI benchmarks (UCI Machine Learning Repository, 2014; Brooks et al., 1989; Rana, 2013); California housing follows Pace & Barry (1997); CPU activity, Kin8nm, and Pumadyn32nh are DELVE benchmarks (DELVE, 1996a;b;c); Friedman2 follows Friedman (1991). The same RAL search ranges, support rules, correction thresholds, risk procedure, and conformal settings are used throughout; the complete frozen configuration is given in Appendix A.

We distinguish two comparison settings. *Standalone* models replace the original predictor and include linear regression, EBM (Nori et al., 2019), random forest (Breiman, 2001), extremely randomized trees (Geurts et al., 2006), XGBoost (Chen & Guestrin, 2016), and LightGBM (Ke et al., 2017). *Fixed-base* correctors preserve the same proper-training subset and base specification as RAL and construct residual targets using the same five-fold OOF procedure, with independently seeded OOF assignments. They include a shallow tree, EBM, KNN, gradient boosting, random forest, extremely randomized trees, XGBoost, and LightGBM. We additionally evaluate ungated, hard-gated, and no-local-affine RAL ablations.

MAE is the primary metric. Cross-dataset comparisons additionally use

$$\mathrm{NMAE}_{\mathrm{SD}} = \frac{\mathrm{MAE}}{\mathrm{SD}(y_{\mathrm{train}})}, \tag{11}$$

with a common denominator for all methods within each outer fold. We also report RMSE, tail absolute error, maximum absolute error, and R^2. Outer folds are first averaged within dataset; statistical tests therefore use the eight dataset-level means as paired units rather than treating 40 folds as independent replicates.

To quantify how much correction signal survives the analytical constraint, we report

$$\eta_D = \frac{\mathrm{MAE}_{\mathrm{base},D} - \mathrm{MAE}_{\mathrm{RAL},D}}{\mathrm{MAE}_{\mathrm{base},D} - \min_{m\in\mathcal{M}} \mathrm{MAE}_{m,D}}, \tag{12}$$

where $\mathcal{M}$ denotes the tested residual correctors. η_D is descriptive, not an optimality claim.

Finally, post-correction risk is evaluated through risk–error association, risk groups, and selective prediction at retained coverages of 90%, 80%, 70%, 50%, and 30%, against random and oracle retention. Global and risk-normalized conformal intervals are evaluated at 80%, 90%, 95%, and 99% nominal coverage using empirical coverage, mean width, and interval score.

3 Results

The results reveal a recurring separation between the structure needed to make a prediction and the structure needed to correct its remaining error. We first examine this residual representation, then its predictive value and auditability, and finally whether the error remaining after correction can itself be predicted.

3.1 Residual Structure and Stability

Pumadyn32nh provides the clearest example. Its frozen linear base uses all 32 predictors, yet RAL selects `tau4` and `theta5` in all five outer folds, with these two variables forming the complete residual subspace in four. Despite this compression, RAL reduces base MAE by 33.1%.

Figure 1a shows the same pattern across datasets: selected residual representations contain only two to four variables, including three of 21 for CPU activity and two of eight for California. The selected variables are also recurrent across independently repeated outer folds: Friedman2 repeatedly selects x_2, x_3, California longitude and latitude, Airfoil frequency and chord, and CPU activity `freeswap` and `runqsz`. Full fold-level selections are reported in Appendix C.

Kin8nm is an important qualification: the same four-variable set is selected in all five folds, but $d = 4$ is also the search limit. Its dimension should therefore be interpreted as right-censored. More generally, these results do not imply that other variables are unimportant to the target; they remain in the frozen base. Rather, useful residual correction repeatedly concentrates in a much smaller conditional representation.

3.2 Residual Correction Performance

The compact representation remains useful for prediction. Table 2 shows that RAL lowers base MAE on all eight datasets and all 40 outer folds. Dataset-level reductions range from 9.6% to 97.0%,

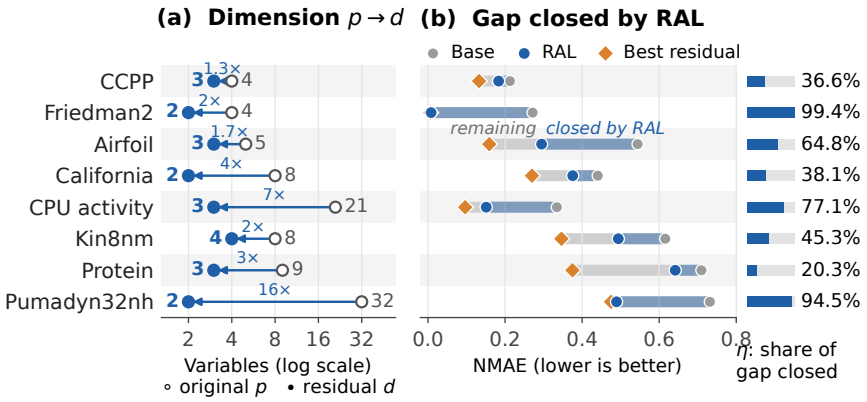

Figure 1: Compact residual structure and retained correction. (a) Original dimension p versus median selected residual dimension d. (b) Frozen-base, RAL, and best tested residual-corrector NMAE; η_D denotes the fraction of the best observed correction improvement recovered by RAL.

Table 2: Fixed-base residual correction. ΔMAE is reduction relative to the frozen base; Best is the lowest tested residual-corrector NMAE, and η_D is the fraction of its improvement recovered by RAL.

Dataset	Base	RAL	ΔMAE	Best NMAE	Method	η_D
CCPP	0.2126	0.1833	13.8%	0.1326	RF	36.6%
Friedman2	0.2715	0.0080	97.0%	0.0065	ET	99.4%
Airfoil	0.5441	0.2948	45.8%	0.1591	ET	64.8%
California	0.4406	0.3755	14.8%	0.2699	RF	38.1%
CPU activity	0.3345	0.1511	54.8%	0.0965	EBM	77.1%
Kin8nm	0.6158	0.4939	19.8%	0.3464	KNN	45.3%
Protein	0.7097	0.6417	9.6%	0.3750	ET	20.3%
Pumadyn32nh	0.7316	0.4893	33.1%	0.4753	EBM	94.5%
Mean	–	–	36.1%	–	–	59.5%
Median	–	–	26.5%	–	–	55.0%

with mean 36.1% and median 26.5%. An exact paired Wilcoxon test on the eight dataset-level NMAE means gives $p = 0.0078$ two-sided ($p = 0.0039$ directional). Fold-level results are given in Appendix D.

Flexible residual learners remain more accurate on most datasets: RAL has average NMAE rank 5.88 among nine correctors, and EBM is more accurate on seven of eight. The Friedman test rejects equal ranks ($\chi^2 \approx 39.47$, $p = 4.03 \times 10^{-6}$); complete results appear in Appendix F. The relevant trade-off is therefore how much correction signal survives the analytical constraint. RAL recovers a median $\eta_D = 55.0\%$, reaching 94.5% on Pumadyn32nh despite its essentially two-variable residual representation.

The gains extend beyond MAE: RMSE and R^2 improve on all eight datasets. Tail improvements are less uniform, with P95 improving on seven datasets and P99 on six. Protein and California provide the main exceptions, showing that lower average error does not guarantee improvement throughout the error distribution. Full metrics are reported in Appendix E.

3.3 Instance-Level Auditability

The value of the analytical representation is that the exposed rule is the rule that actually changes the prediction. Figure 2 shows one CPU activity example: a frozen prediction of 84.423 is associated with a supported `freeswap/scall` region, for which the displayed local-affine law produces correction +8.098 and final prediction 92.521.

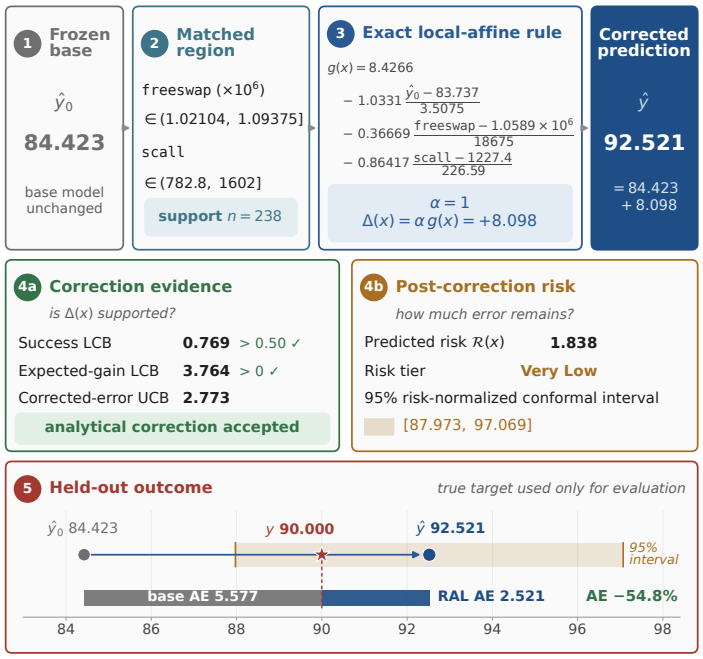

Figure 2: Auditable correction example. A frozen CPU prediction is corrected by the exact displayed analytical rule. Correction evidence and post-correction risk are distinct outputs; the true target is used only for held-out evaluation.

Table 3: Main RAL ablations. No local affine reports the MAE increase after removing that family and retraining.

Dataset	No local affine	Strict < ungated?
CCPP	+4.0%	No
Friedman2	+318.7%	Yes
Airfoil	+27.7%	No
California	+2.7%	No
CPU activity	+18.2%	Yes
Kin8nm	+7.8%	Yes
Protein	+0.5%	No
Pumadyn32nh	+1.6%	No

The same certificate records why the correction was accepted and what uncertainty remains: success LCB 0.769, expected-gain LCB 3.764, corrected-error UCB 2.773, predicted absolute-error risk 1.838, and a 95% risk-normalized conformal interval $[87.973, 97.069]$. The true held-out target, used only retrospectively, is 90.000; absolute error falls from 5.577 to 2.521, a 54.8% reduction.

Thus the displayed law is not a surrogate explanation: it is the computation that generated the adjustment.

3.4 Ablation Analysis

Removing the local-affine family and retraining worsens MAE on all eight datasets, from 0.5% on Protein to 318.7% on Friedman2 ($p = 0.0078$ on dataset-level NMAE). Its heterogeneous effect supports a small library of analytical laws rather than a single correction family.

Held-out strict selection serves a different purpose. It improves MAE over the ungated policy on only three datasets and the paired difference is not significant ($p = 0.5469$). On Airfoil it slightly worsens MAE but improves P99 and maximum error. We therefore interpret strict gating as a conservative

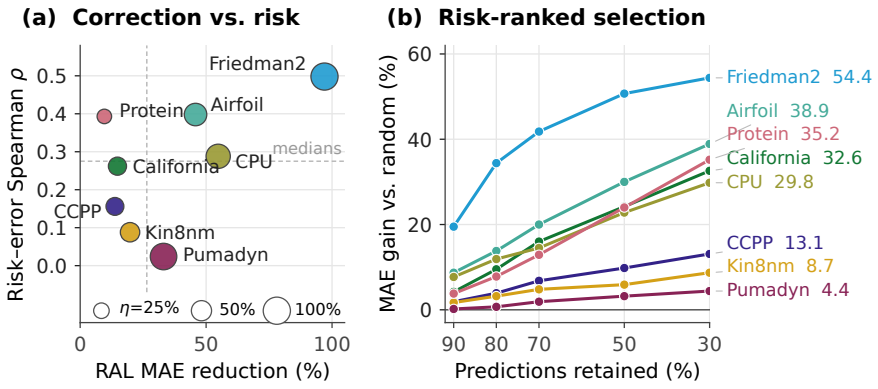

Figure 3: Correction and remaining-error risk. (a) RAL MAE reduction versus risk–error rank correlation; bubble area is proportional to η_D. (b) MAE gain from risk-ranked retention relative to random retention across five retained fractions.

deployment mechanism rather than a general source of average accuracy. Complete ablations appear in Appendix H.

3.5 Post-Correction Risk

Correction learnability and remaining-error predictability can differ sharply. Pumadyn32nh reduces base MAE by 33.1% and recovers 94.5% of the best tested correction gain, yet its mean risk–error Spearman correlation is only 0.024. By contrast, it reaches 0.498 on Friedman2, 0.398 on Airfoil, 0.393 on Protein, 0.288 on CPU activity, and 0.262 on California. Figure 3a summarizes this separation.

Correction-benefit evidence should therefore not be interpreted as general prediction confidence. Its AUROC ranges only from approximately 0.524 to 0.592, with Brier skill often near zero or negative. It is used as conservative evidence for applying a correction, while the separate risk model targets remaining absolute error. Full diagnostics appear in Appendix I.

3.6 Selective Prediction and Conformal Inference

Despite heterogeneous risk ranking, retaining the lowest-risk predictions outperforms random retention in all 40 dataset–coverage comparisons (Figure 3b). At 30% retained coverage, MAE gain ranges from 4.4% on Pumadyn32nh to 54.4% on Friedman2, with mean 27.2%, median 31.2%, and paired $p = 0.0078$. Full selective results appear in Appendix J.

Risk normalization also affects conformal efficiency heterogeneously. Both global and risk-normalized intervals remain close to nominal marginal coverage, but at 95% coverage risk normalization narrows intervals on six of eight datasets and improves interval score on five. It worsens both measures on Pumadyn32nh, consistent with its weak risk ranking. Complete conformal results appear in Appendix K.

Pumadyn32nh therefore closes the empirical story: its systematic residual error is strongly and compactly correctable, while its remaining error is poorly ranked. The experiments consequently separate three questions: whether systematic error can be corrected, whether that correction can be expressed analytically, and whether the error remaining afterward can itself be predicted.

4 Discussion

The central implication of our results is that the error of a fitted predictor can constitute a different learning problem from the original target. Once the base model has absorbed part of the structure in the data, the remaining systematic error is conditional on that model and may admit a substantially more compact representation. RAL exploits this distinction rather than attempting to relearn the

original regression function. The selected residual variables should therefore be interpreted as coordinates of model failure, not as a second feature-importance ranking for the target. This also limits the claim: our results do not establish that residuals are generally simpler than prediction problems, but they show that useful correction structure can be compact even when the original predictor is not.

This perspective changes the role of interpretability. RAL does not attempt to make the entire prediction function simple. Instead, it preserves the fitted predictor and constrains only the intervention made on its output. Auditability therefore attaches to the correction itself: the variables, region, analytical law, and numerical adjustment responsible for changing a prediction are the actual computation used at inference time. This is narrower than claiming global interpretability of the full predictive system, but it is also more direct than explaining a flexible correction model after the fact. The accuracy gap to unrestricted residual learners is consequently not an anomaly to be hidden; it is the cost of constraining the correction to remain analytically recoverable. The relevant empirical question is how much useful correction survives that constraint.

A second implication is that learning systematic error and learning the error that remains afterward are not the same problem. A correction can be reliable in aggregate without making the final error easy to rank at the instance level. This separation argues against collapsing correction evidence and prediction uncertainty into a single confidence score. In RAL, the former determines whether there is sufficient evidence to modify the frozen prediction, whereas the latter describes the expected magnitude of the error after that decision has been made. The selective-prediction and conformal experiments further show that adaptive uncertainty mechanisms are useful only to the extent that this remaining error contains learnable heterogeneity. Marginal coverage can still be calibrated when local risk is weak, but local risk normalization need not improve efficiency.

Together, these observations suggest a model-preserving alternative to the usual choice between retaining an existing predictor and replacing it with a more flexible one. When systematic failure structure can be isolated, an existing model can instead be augmented by an explicit residual layer whose effect is independently validated and whose intervention remains auditable. The broader opportunity is therefore not simply post-hoc residual correction, but treating the failure structure of an already-trained model as a learning object in its own right. Whether the same structure persists for stronger base learners, different data regimes, and distribution shift is an important question for future work.

5 Conclusion

This work studies the failure structure of an already-trained regressor as a learning problem in its own right. Rather than relearning the target, Residual Analytical Learning (RAL) searches for structure in the errors left by a frozen predictor and represents the resulting correction with explicit analytical laws. Across eight heterogeneous benchmarks, the selected residual subspaces have median dimension two to four under a search capped at four variables, with seven of eight using at most three, while improving the frozen base in all 40 outer folds. Flexible residual learners remain more accurate on most datasets, but RAL shows that a substantial part of the available correction signal can survive the constraint that the correction itself remain exactly inspectable. The experiments also expose an important separation: learning systematic error does not imply learning the error that remains after correction. This motivates treating correction selection and post-correction risk as distinct problems rather than collapsing them into a single confidence score. Taken together, the results suggest a model-preserving view of regression improvement: keep the existing predictor, learn a compact and auditable representation of its systematic failures when such structure exists, and model the remaining uncertainty separately. More broadly, they point to model failure itself—not only the original prediction task, as a potentially useful object of representation learning.

AI use statement

Generative AI tools were used during this work to assist with manuscript organization and language editing, LaTeX formatting, and debugging and visualization of research code. AI-assisted code and analyses were reviewed and validated by the authors before use, and the reported experimental results were produced and checked using the authors' executable experimental pipeline. Generative

AI was not used to fabricate data, results, citations, or experimental evidence. The authors take full responsibility for the final methodology, experiments, claims, code, figures, and text in this submission.

Ethics statement

This work is methodological and evaluates regression algorithms on established benchmark datasets. It does not involve human participants, intervention on human subjects, or the collection or release of personally identifiable information. RAL provides instance-level correction evidence and post-correction risk estimates, but these quantities should not be interpreted as guarantees of correctness or safety. We are not aware of additional ethical concerns specific to the experiments reported here.

Reproducibility statement

We provide the complete implementation used for the reported experiments as anonymous supplementary source code accompanying this submission. The main text specifies the learning and evaluation protocol, while Appendix A reports the frozen configuration and comparator settings and Appendix B gives the complete training and inference procedures. Additional appendices report fold-level feature selection, prediction results, ablations, risk diagnostics, selective prediction, conformal evaluation, and statistical summaries. The supplementary code includes the experimental pipeline required to reproduce these analyses. To preserve double-blind review, no identifying public repository is linked in the submitted version. Following the anonymous review process, the code and supporting materials are intended to be released publicly through GitHub and archived on Zenodo.

References

Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey Hinton. Neural additive models: Interpretable machine learning with neural nets. In *Advances in Neural Information Processing Systems*, volume 34, pp. 4699–4711, 2021.

Hubert Baniecki and Przemyslaw Biecek. Adversarial attacks and defenses in explainable artificial intelligence: A survey. *Information Fusion*, pp. 102303, 2024.

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

Thomas Brooks, D. Pope, and Michael Marcolini. Airfoil self-noise. UCI Machine Learning Repository, 1989.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016. doi: 10.1145/2939672.2939785.

William S Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74(368):829–836, 1979.

DELVE. Delve: Computer activity regression benchmark. Data for Evaluating Learning in Valid Experiments, 1996a. URL `https://www.cs.toronto.edu/~delve/data/comp-activ/desc.html`. Accessed 2026.

DELVE. Delve: Kin family regression benchmarks. Data for Evaluating Learning in Valid Experiments, 1996b. URL `https://www.cs.toronto.edu/~delve/data/kin/desc.html`. Accessed 2026.

DELVE. Delve: Puma dynamics regression benchmarks. Data for Evaluating Learning in Valid Experiments, 1996c. URL `https://www.cs.toronto.edu/~delve/data/pumadyn/desc.html`. Accessed 2026.

Jerome H. Friedman. Multivariate adaptive regression splines. *The Annals of Statistics*, 19(1):1–67, 1991. doi: 10.1214/aos/1176347963.

Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001. doi: 10.1214/aos/1013203451.

Yonatan Geifman and Ran El-Yaniv. Selectivenet: A deep neural network with an integrated reject option. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2151–2159, 2019.

Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine Learning*, 63(1):3–42, 2006. doi: 10.1007/s10994-006-6226-1.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In *Advances in Neural Information Processing Systems*, volume 35, pp. 507–520, 2022.

Trevor J. Hastie and Robert J. Tibshirani. *Generalized Additive Models*, volume 43 of *Monographs on Statistics and Applied Probability*. Chapman and Hall/CRC, London, 1990. ISBN 978-0412343902.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30, pp. 3146–3154, 2017.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523): 1094–1111, 2018. doi: 10.1080/01621459.2017.1307116.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Christoph Molnar. *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*. Self-published, 2 edition, 2022. `https://christophm.github.io/interpretable-ml-book/`.

Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. InterpretML: A unified framework for machine learning interpretability. *arXiv preprint arXiv:1909.09223*, 2019.

R. Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33 (3):291–297, 1997. doi: 10.1016/S0167-7152(96)00140-X.

John C Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3):61–74, 1999.

Prashant Rana. Physicochemical properties of protein tertiary structure. UCI Machine Learning Repository, 2013.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144, 2016. doi: 10.1145/2939672.2939778.

Yaniv Romano, Evan Patterson, and Emmanuel J. Candès. Conformalized quantile regression. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

UCI Machine Learning Repository. Combined cycle power plant. UCI Machine Learning Repository, 2014. URL `https://archive.ics.uci.edu/dataset/294/combined+cycle+power+plant`.

Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 694–699, 2002.

APPENDIX

A FROZEN CONFIGURATION AND REPRODUCIBILITY

All eight datasets are evaluated with the same frozen methodological configuration. Dataset-specific quantities such as file paths, target names, and feature names differ, but no search range, support threshold, correction threshold, risk rule, or evaluation hyperparameter is changed in response to outer-test performance. The complete configuration used for the reported experiments is summarized in Tables **??**, **??**, and **??**.

In all 40 outer folds, the automatic base-model rule selected ordinary linear regression rather than Ridge regression. The Ridge option nevertheless remains part of the frozen implementation and is triggered only by the predefined dimensionality or conditioning criteria reported below.

Fixed-base comparator construction. Every residual-correction comparator uses the same proper-training subset and the same base-model specification as RAL, and constructs its residual target using the same five-fold OOF procedure. In the current implementation these baseline OOF predictions use a separate deterministic seed offset, so the OOF fold assignments, and therefore the numerical residual vectors, need not be identical to those used internally by RAL. The comparison nevertheless fixes the data subset, base class, OOF construction protocol, and outer-test fold.

A.1 ADDITIONAL IMPLEMENTATION DETAILS

Residual screening and regional support. Feature screening is performed on proper-training OOF residuals. Stability screening uses five folds and mutual information, and variables participating in strong pairwise residual interactions may be promoted into the candidate pool. For a candidate residual subset, each selected variable is partitioned by empirical quantiles. A fine region is treated as unsupported when it contains fewer than 15 observations; candidate configurations are rejected when their median fine-region support is below 20 or when more than 35% of search observations fall in unsupported fine regions.

Analytical law families. At each supported lattice node, the constant law predicts the median residual. The linear and hinge families model residuals as explicit functions of the base prediction $\hat{y}_0$. The monotone-piecewise family fits both increasing and decreasing isotonic relations and retains the lower-error direction, represented using at most five piecewise-linear knots. The local-affine family uses the standardized base prediction together with the variables defining the corresponding subset, with ridge penalty 10^{-4}. Law family and shrinkage are chosen from cross-fitted proper-training predictions before the final law is refit on all proper-training observations in the node.

Extrapolation control. For a deployed law, the base prediction and any feature values entering that law are clipped to the ranges observed during fitting, and the resulting correction is clipped to the 1st–99th percentiles of the corresponding proper-training residuals. Let $d_{\mathrm{ext}}(x)$ denote the normalized distance of an input outside the fitted range. The extrapolation evidence factor is

$$p_{\mathrm{ext}}(x) = \exp\{-4 d_{\mathrm{ext}}(x)\}. \tag{13}$$

This factor modifies the held-out evidence used for candidate selection; it is not multiplied into the analytical correction itself.

Correction-calibration backoff. Held-out correction evidence is first retrieved for the exact region–subset–family state. When exact support is insufficient, evidence is pooled successively by subset size and law family, by law family, and then by correction source/global evidence. Pooled correction states require at least 50 calibration observations. Analytical candidates must have an adjusted success lower bound above 0.50 and a positive adjusted gain lower bound; the separately trained constant fallback uses the stricter success threshold 0.55.

Post-correction risk backoff. Risk estimation is performed only after the correction policy is frozen. An exact risk state is determined by correction source, selected subset, region, and law family

and is used when it contains at least 15 calibration observations. Otherwise RAL backs off through pooled states based on source/subset-size and law family, then broader family/source states, and finally the complete calibration set; pooled states require at least 40 observations. For each candidate blend $w \in \{0, 0.1, \ldots, 1\}$, predicted absolute-error risk is compared with realized corrected absolute error on the correction/risk-calibration subset, and the value minimizing mean absolute deviation is retained.

Finite-sample conformal quantile. After the correction and risk policies are frozen, normalized conformity scores are computed only on the untouched conformal-calibration subset. The standard finite-sample split-conformal quantile is used at each requested nominal coverage; no test observation or outer-test target enters calibration.

B Complete RAL training and inference procedures

Algorithm 1 expands the condensed description in the main text. Algorithm 2 shows the deployed inference path. Neither algorithm uses outer-test targets during model construction, candidate selection, risk estimation, or conformal calibration.

Algorithm 1 RAL training within one outer-training fold

Require: Outer-training data $\mathcal{D}_{\mathrm{tr}}$

Randomly split $\mathcal{D}_{\mathrm{tr}}$ into proper, correction/risk-calibration, and conformal-calibration subsets with proportions $0.80/0.10/0.10$
Fit median imputation using proper-training features only
Select the base specification by the frozen linear/Ridge rule
Construct five-fold OOF base predictions on proper training
Form OOF residuals $r_i = y_i - \tilde{y}_{0,i}$
Refit the frozen base on the complete proper-training subset

Rank variables by stability-fold mutual information with r
Run the pairwise residual-interaction screen and promote variables participating in strong pairs
Keep at most 12 candidate residual variables

for $d \in \{1, 2, 3, 4\}$ **do**
    for candidate subsets S retained by the beam search **do**
        for $B \in \{3, 4, 5, 6, 7, 8, 10\}$ **do**
            Construct quantile regions over S
            Reject configurations with median fine support < 20 or unsupported-observation rate > 0.35
            Evaluate coarse three-fold OOF regional-median correction including lower-dimensional projection backoff
            Score using corrected MAE plus frozen dimension and region-complexity penalties
        end for
    end for
end for
Keep the 15 strongest coarse configurations
Rerank them by three-fold analytical correction using constant, linear, hinge, and local-affine laws
Select residual subset $S^\star$ and bin count $B^\star$

Build the complete order-independent subset lattice of $S^\star$ plus the global node
for each sufficiently supported lattice region **do**
    Fit candidate constant, linear, hinge, monotone-piecewise, and local-affine laws
    Obtain five-fold OOF predictions for each law
    Select shrinkage $\alpha \in \{0, 0.05, \ldots, 1\}$ by OOF corrected MAE
    Retain the law only if $\alpha > 0$ and OOF gain is positive
end for
Fit the retained laws on the complete proper-training subset
Separately cross-fit and fit the constant-residual fallback policy

Freeze all correction laws
Apply all frozen candidates to the correction/risk-calibration subset
For each candidate state compute correction success, absolute-error gain, and corrected absolute error
Compute one-sided Clopper–Pearson success LCBs, Student-t gain LCBs, and Student-t corrected-error UCBs
Construct exact and pooled correction-calibration states
Freeze the strict analytical selection and constant-fallback policy

Pass the correction/risk-calibration subset through the final policy
Estimate policy-aware final absolute-error risk using exact states and hierarchical pooled backoff
Select blend weight $w \in \{0, 0.1, \ldots, 1\}$ between calibrated state risk and proper-OOF policy risk
Freeze risk estimator and calibration-derived risk-tier cut points

Apply the frozen prediction and risk policies to the untouched conformal-calibration subset
Compute normalized conformity scores $s_i = |y_i - \hat{y}_i| / \max\{\mathcal{R}(x_i), \varepsilon\}$
Store finite-sample conformal quantiles for $1 - \alpha \in \{0.80, 0.90, 0.95, 0.99\}$
return frozen RAL model

Algorithm 2 RAL prediction for a new input

Require: Frozen RAL model and input x
Apply the proper-training median imputer
Compute frozen base prediction $\hat{y}_0 = f_0(x)$
Enumerate supported analytical candidates across the subset lattice
for each candidate **do**
    Identify its quantile region
    Clip law inputs to the ranges observed during law fitting
    Evaluate the fitted analytical law and its learned shrinkage
    Clip the resulting correction to the training residual 1st–99th percentile range
    Compute extrapolation distance and corresponding evidence penalty
    Retrieve exact or pooled held-out correction statistics
end for
Keep analytical candidates whose adjusted success LCB > 0.50 and adjusted gain LCB > 0
if at least one analytical candidate passes **then**
    Select the candidate with the smallest calibrated corrected-error UCB
    Break ties by larger success LCB and then more specific subset
else if a constant-residual fallback passes success LCB > 0.55 and gain LCB > 0 **then**
    Apply the selected constant-residual fallback
else
    Set $\Delta(x) = 0$ and retain the base prediction
end if
Compute final prediction $\hat{y} = \hat{y}_0 + \Delta(x)$
Retrieve exact or pooled policy-aware state risk
Blend state risk with the corresponding proper-OOF policy risk
Assign predicted-risk tier from calibration-derived cut points
Construct conformal interval $I_\alpha(x) = [\hat{y} - q_{1-\alpha}\mathcal{R}(x), \hat{y} + q_{1-\alpha}\mathcal{R}(x)]$
return correction certificate, final prediction, risk, and interval

C RESIDUAL REPRESENTATION STABILITY

Table 7 gives the selected residual coordinates and quantile-bin count in every outer fold. The order in which variables are printed is not meaningful because deployment uses an order-independent subset lattice.

Table 7: Selected residual variables and quantile-bin count B by outer fold.

(a) Folds 1–3

Dataset	Fold 1	Fold 2	Fold 3
CCPP	V, AT, AP ($B = 5$)	V, AP ($B = 10$)	AT, V, AP ($B = 6$)
Friedman2	x_3, x_2 ($B = 10$)	x_2, x_3 ($B = 10$)	x_3, x_2 ($B = 10$)
Airfoil	freq, disp, chord ($B = 3$)	freq, angle, chord ($B = 3$)	freq, chord, angle ($B = 3$)
California	longitude, latitude, income ($B = 7$)	longitude, latitude ($B = 10$)	longitude, latitude ($B = 10$)
CPU activity	freeswap, runqsz, scall ($B = 6$)	freeswap, runqsz, vflt ($B = 6$)	freeswap, runqsz, scall ($B = 6$)
Kin8nm	$\theta_5, \theta_6, \theta_4, \theta_7$ ($B = 3$)	$\theta_4, \theta_6, \theta_5, \theta_7$ ($B = 3$)	$\theta_5, \theta_4, \theta_7, \theta_6$ ($B = 3$)
Protein	F3, F6, F8 ($B = 6$)	F4, F3, F8 ($B = 5$)	F4, F3, F8 ($B = 5$)
Pumadyn32nh	tau4, theta5 ($B = 6$)	tau4, theta5 ($B = 5$)	tau4, theta5 ($B = 6$)

(b) Folds 4–5

Dataset	Fold 4	Fold 5
CCPP	V, RH, AP ($B = 5$)	AT, V, AP ($B = 5$)
Friedman2	x_3, x_2 ($B = 10$)	x_2, x_3 ($B = 10$)
Airfoil	freq, chord ($B = 4$)	freq, chord, angle ($B = 3$)
California	longitude, latitude ($B = 10$)	longitude, latitude ($B = 10$)
CPU activity	freeswap, runqsz, vflt ($B = 6$)	freeswap, scall, runqsz ($B = 5$)
Kin8nm	$\theta_5, \theta_7, \theta_6, \theta_4$ ($B = 3$)	$\theta_5, \theta_4, \theta_7, \theta_6$ ($B = 3$)
Protein	F4, F3, F8 ($B = 5$)	F4, F3, F8 ($B = 5$)
Pumadyn32nh	tau4, theta5, thetad3 ($B = 6$)	tau4, theta5 ($B = 6$)

Table 8: Percentage MAE reduction of strict RAL relative to its frozen base in each outer fold. Positive values indicate improvement.

Dataset	F1	F2	F3	F4	F5
CCPP	14.7%	13.7%	15.2%	13.4%	11.7%
Friedman2	97.4%	96.6%	97.4%	96.9%	97.0%
Airfoil	50.5%	46.1%	48.5%	37.7%	46.5%
California	18.8%	13.2%	14.3%	13.6%	14.0%
CPU activity	53.7%	55.8%	54.6%	56.9%	53.1%
Kin8nm	21.1%	20.8%	19.3%	18.8%	19.1%
Protein	12.3%	8.6%	9.0%	8.7%	9.3%
Pumadyn32nh	33.3%	29.4%	31.9%	37.3%	33.4%

Pumadyn32nh therefore contains the same two-variable core (`tau4`, `theta5`) in all five folds, with only Fold 4 adding `thetad3`. Kin8nm selects the same four-variable set in all five folds, but this result is right-censored because four is also the maximum permitted residual dimension.

D Fold-level improvement over the frozen base

The main text states that RAL improves the frozen base on all 40 outer folds. Table 8 reports the corresponding fold-level MAE reductions explicitly.

E Extended point-prediction results

Table 9 expands the main MAE/NMAE comparison to RMSE, tail absolute error, and R^2. Percentage changes for error metrics are computed as $(\text{base} - \text{RAL})/\text{base}$, so positive values indicate improvement.

Table 9: Extended comparison of the frozen base and strict RAL. Positive percentage changes denote error reduction; negative values denote deterioration.

(a) MAE and RMSE

	MAE			RMSE		
Dataset	Base	RAL	Δ	Base	RAL	Δ
CCPP	3.6277	3.1283	13.8%	4.5596	4.1018	10.0%
Friedman2	103.47	3.0561	97.0%	137.11	5.0589	96.3%
Airfoil	3.7521	2.0327	45.8%	4.8310	2.7628	42.8%
California	50,837.5	43,329.7	14.8%	69,629.7	64,582.3	7.2%
CPU activity	6.1532	2.7785	54.8%	9.6797	4.0335	58.3%
Kin8nm	0.1623	0.1302	19.8%	0.2020	0.1648	18.4%
Protein	4.3421	3.9262	9.6%	5.1857	5.1412	0.9%
Pumadyn32nh	0.0263	0.0176	33.1%	0.0330	0.0222	32.8%

(b) P95 and P99 absolute error

	P95 absolute error			P99 absolute error		
Dataset	Base	RAL	Δ	Base	RAL	Δ
CCPP	8.3070	7.7074	7.2%	11.375	11.188	1.6%
Friedman2	285.10	10.924	96.2%	349.18	19.371	94.5%
Airfoil	9.7992	5.7844	41.0%	13.412	8.4052	37.3%
California	143,558.8	139,262.0	3.0%	231,191.2	237,079.7	-2.5%
CPU activity	13.830	7.8565	43.2%	44.994	12.652	71.9%
Kin8nm	0.3898	0.3246	16.7%	0.5345	0.4506	15.7%
Protein	9.5672	10.477	-9.5%	12.657	13.810	-9.1%
Pumadyn32nh	0.0652	0.0441	32.3%	0.0825	0.0576	30.2%

(c) R^2

Dataset	Base	RAL
CCPP	0.929	0.942
Friedman2	0.870	1.000
Airfoil	0.505	0.837
California	0.636	0.686
CPU activity	0.722	0.951
Kin8nm	0.413	0.609
Protein	0.281	0.294
Pumadyn32nh	0.155	0.618

The extended metrics show that MAE, RMSE, and R^2 improve on every dataset. Tail error is less uniform: P95 improves on seven of eight datasets and P99 on six. Protein has lower average error but higher P95 and P99 absolute error, while California exhibits a small P99 deterioration.

F Complete fixed-base residual-correction benchmark

Table 10 reports NMAE for every residual corrector present in the exported benchmark summaries. Lower is better. The final row reports the average rank across the eight datasets.

The Friedman test across the nine residual-correction methods gives $\chi^2 = 39.47$ with $p = 4.03 \times 10^{-6}$.

Table 10: NMAE of all tested fixed-base residual-correction methods. Bold marks the lowest NMAE in each dataset. Tree denotes the shallow residual tree and GB denotes gradient boosting.

(a) RAL, EBM, and compact residual baselines

Dataset	RAL	EBM	Tree	KNN	GB
CCPP	0.1833	0.1459	0.1969	0.1661	0.1857
Friedman2	0.0080	0.0080	0.2253	0.0314	0.1081
Airfoil	0.2948	0.2078	0.4185	0.3856	0.3938
California	0.3755	0.2907	0.4186	0.3523	0.3807
CPU activity	0.1511	**0.0965**	0.1842	0.1458	0.1504
Kin8nm	0.4939	0.5034	0.6037	**0.3464**	0.5718
Protein	0.6417	0.5292	0.6749	0.4339	0.6348
Pumadyn32nh	0.4893	**0.4753**	0.7116	0.7405	0.7170
Average rank	5.88	4.00	8.75	5.62	7.75

(b) Flexible tree-based residual baselines

Dataset	RF	ET	XGB	LGB
CCPP	**0.1326**	0.1327	0.1386	0.1394
Friedman2	0.0094	**0.0065**	0.0241	0.0128
Airfoil	0.2018	**0.1591**	0.1705	0.1805
California	**0.2699**	0.2778	0.2852	0.2768
CPU activity	0.1071	0.1015	0.0989	0.1025
Kin8nm	0.3955	0.3636	0.3848	0.3635
Protein	0.4004	**0.3750**	0.4897	0.4940
Pumadyn32nh	0.5080	0.5696	0.5400	0.5364
Average rank	3.12	2.50	3.75	3.62

Table 11: Standalone-model NMAE. Lower is better. These models use the complete outer-training fold and are not direct fixed-base correction competitors.

Dataset	LR	EBM	RF	ET	XGB	LGB
CCPP	0.2126	0.1419	0.1393	0.1376	**0.1366**	0.1388
Friedman2	0.2715	0.0102	0.0108	**0.0064**	0.0256	0.0120
Airfoil	0.5444	0.1899	0.2145	**0.1506**	0.1606	0.1768
California	0.4402	0.2819	0.2737	0.3013	**0.2687**	0.2707
CPU activity	0.3299	0.0914	0.0923	0.0881	**0.0832**	0.0838
Kin8nm	0.6155	0.5023	0.4204	0.4046	0.3701	**0.3500**
Protein	0.7096	0.5206	0.3925	**0.3672**	0.4758	0.4839
Pumadyn32nh	0.7307	0.4739	0.4785	**0.4736**	0.4849	0.4823

G STANDALONE PREDICTIVE BENCHMARK

The standalone benchmark answers a different question from the fixed-base benchmark: how much accuracy is available if the original predictor is replaced rather than preserved. Table 11 reports dataset-level NMAE for the standalone models present in all exported summaries.

H ABLATION RESULTS

Table 12 reports the complete dataset-level MAE comparison for the four RAL policies exported by the experiment. "No local affine" repeats the complete training pipeline after removing the local-affine family, rather than merely disabling that law at inference.

Removing local affine worsens MAE on all eight datasets, but its effect varies from negligible on Protein to dominant on Friedman2. Strict selection is more accurate than ungated selection on only

Table 12: Dataset-level MAE for the principal RAL ablations. The final column is the relative MAE increase of the no-local-affine retraining relative to strict RAL.

Dataset	Base only	Strict	Ungated	Hard-gated	No local affine	No-local increase
CCPP	3.6277	3.1283	3.0695	3.1986	3.2522	4.0%
Friedman2	103.47	3.0561	12.972	3.0921	12.795	318.7%
Airfoil	3.7521	2.0327	2.0247	2.1091	2.5963	27.7%
California	50,837.5	43,329.7	39,726.3	42,759.2	44,502.3	2.7%
CPU activity	6.1532	2.7785	3.2270	3.2113	3.2850	18.2%
Kin8nm	0.1623	0.1302	0.1309	0.1307	0.1404	7.8%
Protein	4.3421	3.9262	3.6197	3.8426	3.9443	0.5%
Pumadyn32nh	0.0263	0.0176	0.0173	0.0194	0.0179	1.6%

Table 13: Correction-benefit and final-risk diagnostics averaged over the five outer folds. Accept rate is the fraction receiving a non-base correction. "Harm among accepted" is the realized fraction of accepted corrections that increase absolute error on the outer-test observations.

Dataset	AUROC	Brier skill	Risk–error ρ	Accept rate	Base fallback	Harm among accepted
CCPP	0.524	-0.006	0.156	100.0%	0.0%	39.3%
Friedman2	0.533	0.012	0.498	100.0%	0.0%	2.7%
Airfoil	0.537	-0.014	0.398	99.9%	0.1%	27.8%
California	0.587	0.017	0.262	99.995%	0.005%	38.0%
CPU activity	0.592	0.018	0.288	100.0%	0.0%	27.1%
Kin8nm	0.530	0.002	0.088	99.988%	0.012%	37.6%
Protein	0.588	0.014	0.393	100.0%	0.0%	38.3%
Pumadyn32nh	0.539	-0.005	0.024	87.8%	12.2%	32.5%

three datasets (Friedman2, CPU activity, and Kin8nm), consistent with the interpretation of strict gating as a deployment regularizer rather than a universally accuracy-maximizing rule.

I Correction-benefit and post-correction risk diagnostics

Table 13 reports fold-averaged correction-benefit and final-risk diagnostics. AUROC and Brier skill concern prediction of whether an accepted correction reduces absolute error relative to the base. Spearman ρ concerns the separate post-correction risk estimator.

The correction-benefit classifier is weak to moderate even on datasets where the average correction is highly effective. This is why its output is described as conservative evidence for applying a correction rather than as a probability that the final prediction is correct.

I.1 Calibration-derived predicted-risk groups

The implementation exports the historical labels "Very High", "High", "Medium", and "Low" certainty. In the paper we invert these labels to the more precise predicted-risk terminology:

$$\text{Very High certainty} \to \text{Very Low risk}, \qquad \text{High certainty} \to \text{Low risk},$$
$$\text{Medium certainty} \to \text{Moderate risk}, \qquad \text{Low certainty} \to \text{High risk}.$$

Table 14 reports pooled outer-test MAE using the paper-facing terminology.

Seven datasets exhibit monotonically increasing realized MAE from Very Low to High predicted risk. Pumadyn32nh is the exception and is also the dataset with the weakest fold-level risk–error rank correlation.

Table 14: Actual MAE by predicted-risk group. "Ratio" is the MAE of the highest predicted-risk group divided by the MAE of the lowest.

Dataset	Very Low	Low	Moderate	High	Monotone?	Ratio
CCPP	2.7240	2.9934	3.3512	3.6569	Yes	1.34×
Friedman2	1.3814	1.6393	2.6386	6.5533	Yes	4.74×
Airfoil	1.1813	1.6611	2.3567	3.0962	Yes	2.62×
California	28,406.3	38,026.2	53,350.8	61,077.7	Yes	2.15×
CPU activity	1.8893	2.4281	2.9850	3.9610	Yes	2.10×
Kin8nm	0.1182	0.1271	0.1318	0.1451	Yes	1.23×
Protein	2.4447	3.6153	4.7067	5.1559	Yes	2.11×
Pumadyn32nh	0.0170	0.0173	0.0184	0.0181	No	1.07×

Table 15: MAE reduction from predicted-risk retention relative to random retention. All 40 dataset–coverage comparisons are positive.

Dataset	90%	80%	70%	50%	30%	Oracle captured at 30%
CCPP	1.9%	3.9%	6.8%	9.8%	13.1%	17.1%
Friedman2	19.5%	34.4%	41.8%	50.7%	54.4%	63.6%
Airfoil	8.7%	13.8%	20.0%	30.0%	38.9%	48.6%
California	4.2%	9.5%	16.0%	24.2%	32.6%	39.9%
CPU activity	7.7%	11.9%	14.5%	22.8%	29.8%	36.5%
Kin8nm	1.7%	3.2%	4.8%	5.9%	8.7%	11.4%
Protein	3.8%	7.8%	12.9%	24.0%	35.2%	42.8%
Pumadyn32nh	0.2%	0.7%	1.9%	3.2%	4.4%	5.8%
Mean	6.0%	10.7%	14.9%	21.3%	27.2%	33.2%
Median	4.0%	8.7%	13.7%	23.4%	31.2%	38.2%

J SELECTIVE PREDICTION

For each retained fraction, predictions are ordered by estimated post-correction risk and the lowest-risk fraction is retained. The random reference is averaged over 100 repeated random subsets at the same retained fraction. Table 15 reports the percentage MAE reduction relative to this random reference.

The final column reports the fraction of the random-to-oracle selective opportunity captured at 30% retained coverage,

$$\kappa_D = \frac{\mathrm{MAE}_{\mathrm{random},D} - \mathrm{MAE}_{\mathrm{risk},D}}{\mathrm{MAE}_{\mathrm{random},D} - \mathrm{MAE}_{\mathrm{oracle},D}}. \tag{14}$$

K COMPLETE CONFORMAL RESULTS

Tables 16–19 report all four nominal coverage levels. "Global" denotes ordinary split conformal using unnormalized absolute residuals. "Local" denotes normalization by the predicted post-correction absolute-error risk. Width and score changes are computed as $(\text{local} - \text{global})/\text{global}$, so negative values indicate improvement.

At 80% and 90% nominal coverage, normalized conformal interval score is lower on seven of eight datasets. At 95% and 99%, it is lower on five of eight. Coverage remains close to nominal for both constructions; efficiency depends on whether the estimated post-correction risk contains useful local error information.

Table 16: Conformal coverage and width for CCPP, Friedman2, Airfoil, and California.

Dataset	Nominal	Global cov.	Local cov.	Global width	Local width
CCPP	80%	0.813	0.808	10.164	10.070
CCPP	90%	0.910	0.913	13.225	13.432
CCPP	95%	0.956	0.954	15.904	16.097
CCPP	99%	0.992	0.993	24.729	25.429
Friedman2	80%	0.794	0.808	8.6302	9.2855
Friedman2	90%	0.902	0.909	15.027	12.706
Friedman2	95%	0.956	0.955	23.443	16.362
Friedman2	99%	0.992	0.991	41.870	26.110
Airfoil	80%	0.821	0.833	7.0605	6.7792
Airfoil	90%	0.912	0.911	9.7885	8.7584
Airfoil	95%	0.954	0.949	12.224	10.855
Airfoil	99%	0.989	0.994	19.835	18.849
California	80%	0.802	0.802	127,434.9	127,747.7
California	90%	0.899	0.901	191,848.0	186,077.0
California	95%	0.950	0.952	281,300.1	257,457.1
California	99%	0.991	0.992	495,885.2	451,783.9

Table 17: Conformal interval score and relative width/score changes for CCPP, Friedman2, Airfoil, and California. Negative changes favor risk normalization.

Dataset	Nominal	Global score	Local score	Width Δ	Score Δ
CCPP	80%	14.222	14.150	-0.9%	-0.5%
CCPP	90%	17.288	17.284	+1.6%	-0.0%
CCPP	95%	20.563	20.628	+1.2%	+0.3%
CCPP	99%	32.365	32.846	+2.8%	+1.5%
Friedman2	80%	18.326	14.631	+7.6%	-20.2%
Friedman2	90%	25.598	18.133	-15.4%	-29.2%
Friedman2	95%	33.498	21.827	-30.2%	-34.8%
Friedman2	99%	51.815	32.659	-37.6%	-37.0%
Airfoil	80%	10.216	9.3719	-4.0%	-8.3%
Airfoil	90%	12.767	11.577	-10.5%	-9.3%
Airfoil	95%	15.400	13.865	-11.2%	-10.0%
Airfoil	99%	22.598	19.765	-5.0%	-12.5%
California	80%	234,426.3	224,444.1	+0.2%	-4.3%
California	90%	314,499.9	293,613.0	-3.0%	-6.6%
California	95%	401,904.9	367,678.9	-8.5%	-8.5%
California	99%	595,178.1	552,194.5	-8.9%	-7.2%

L ADDITIONAL STATISTICAL SUMMARIES

All paired inferential comparisons use the eight dataset-level NMAE means as the statistical units. Outer folds are first aggregated within dataset and are not treated as 40 independent replicates.

Because $n = 8$, exact Wilcoxon tests have coarse attainable p-values. The tests are therefore interpreted together with effect sizes, per-dataset results, and the descriptive accuracy–auditability comparison rather than as standalone evidence.

Table 18: Conformal coverage and width for CPU activity, Kin8nm, Protein, and Pumadyn32nh.

Dataset	Nominal	Global cov.	Local cov.	Global width	Local width
CPU activity	80%	0.806	0.804	8.9901	8.6601
CPU activity	90%	0.903	0.903	12.621	12.110
CPU activity	95%	0.953	0.949	16.134	15.368
CPU activity	99%	0.991	0.992	26.554	25.496
Kin8nm	80%	0.797	0.803	0.4135	0.4187
Kin8nm	90%	0.899	0.897	0.5352	0.5277
Kin8nm	95%	0.947	0.947	0.6436	0.6368
Kin8nm	99%	0.991	0.993	0.9087	0.9250
Protein	80%	0.798	0.801	13.176	12.077
Protein	90%	0.901	0.903	17.432	16.345
Protein	95%	0.951	0.952	21.096	20.803
Protein	99%	0.990	0.990	27.958	34.214
Pumadyn32nh	80%	0.808	0.810	0.0576	0.0590
Pumadyn32nh	90%	0.905	0.904	0.0745	0.0770
Pumadyn32nh	95%	0.950	0.949	0.0883	0.0915
Pumadyn32nh	99%	0.990	0.990	0.1163	0.1238

Table 19: Conformal interval score and relative width/score changes for CPU activity, Kin8nm, Protein, and Pumadyn32nh. Negative changes favor risk normalization.

Dataset	Nominal	Global score	Local score	Width Δ	Score Δ
CPU activity	80%	14.355	13.641	-3.7%	-5.0%
CPU activity	90%	18.312	17.213	-4.0%	-6.0%
CPU activity	95%	22.723	21.073	-4.7%	-7.3%
CPU activity	99%	35.736	32.549	-4.0%	-8.9%
Kin8nm	80%	0.5821	0.5789	+1.3%	-0.5%
Kin8nm	90%	0.6916	0.6870	-1.4%	-0.7%
Kin8nm	95%	0.8018	0.7935	-1.0%	-1.0%
Kin8nm	99%	1.0120	1.0067	+1.8%	-0.5%
Protein	80%	18.573	17.797	-8.3%	-4.2%
Protein	90%	22.017	21.825	-6.2%	-0.9%
Protein	95%	25.074	26.338	-1.4%	+5.0%
Protein	99%	31.249	39.669	+22.4%	+26.9%
Pumadyn32nh	80%	0.0786	0.0796	+2.3%	+1.3%
Pumadyn32nh	90%	0.0931	0.0951	+3.3%	+2.2%
Pumadyn32nh	95%	0.1060	0.1093	+3.5%	+3.1%
Pumadyn32nh	99%	0.1307	0.1395	+6.4%	+6.7%

M TERMINOLOGY AND INTERPRETATION

Several implementation field names predate the final paper terminology. We use the following interpretation consistently in the manuscript.

Correction-benefit estimate. The implementation field `correction_confidence_lcb` is a conservative lower bound associated with evidence that the deployed correction improves absolute error relative to the frozen base. It is not the probability that the final prediction is correct.

Predicted final risk. The implementation field `predicted_abs_error_risk` estimates the magnitude of the final absolute error. It is unsigned and therefore cannot be interpreted as a directional residual correction.

Table 20: Exact paired Wilcoxon tests across the eight datasets. "Wins" counts datasets in which the first-named method has lower error (or, for risk retention, lower error than random retention).

Comparison	Wins	W	Two-sided p
RAL vs. frozen base	8/8	0	0.0078125
RAL vs. no-local-affine retraining	8/8	0	0.0078125
Risk retention vs. random at 30%	8/8	0	0.0078125
Strict vs. ungated	3/8	13	0.546875
RAL vs. shallow-tree residual	8/8	0	0.0078125
RAL vs. EBM residual	1/8	2	0.0234375

Risk groups. The historical implementation labels "Very High/High/Medium/Low certainty" are reported in the paper as "Very Low/Low/Moderate/High predicted risk," respectively. This terminology matches the quantity being estimated and avoids treating risk as a generic notion of correctness.

Residual variables. A variable selected by RAL is interpreted as informative for residual error conditional on the frozen base model. Selection does not imply that the variable is globally more important for predicting the target, nor does it imply a causal relationship.

Local auditability. RAL exposes the exact correction law used for an individual prediction. This does not imply that the complete collection of all possible regional laws is globally small. The auditability claim is therefore instance-level and computational: the numerical adjustment applied to a prediction can be reconstructed exactly from the selected subset, region, law, parameters, and shrinkage.