STRUCTURED ARCHITECTURE PREFERENCE IN TIME-SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Forecasting benchmarks conflate architecture-wide strength, task difficulty, and architecture–task compatibility. We isolate architecture preference as the residual interaction in log forecasting risk after removing the first two effects. Across ten architectures and 41 tasks from eleven datasets, preference is strongly local to dataset identity: residual profiles correlate at 0.800 across horizons of the same dataset but at -0.014 across datasets. When an entire horizon is held out from a previously observed dataset, nearest-horizon reuse recovers 90.8% of held-out pairwise preferences, compared with 43.4% for exchangeable matrix factorization. Matched random-group controls attribute this gain to real dataset membership. The locality pattern reproduces on a separately executed panel of ten datasets and six fixed architectures (within/across $0.722/0.037$; slope -0.306) and is stable across the tested update budgets. Strict leave-one-dataset-out prediction is substantially weaker, showing that observable metadata captures only part of preference on unseen datasets. A matrix-only evaluation-budget audit identifies budget-dependent operational benefits. Residual architecture preference is therefore predominantly dataset-local: global architecture strength transfers broadly, whereas preference is most reusable across related configurations within an observed dataset.

1 INTRODUCTION

Time-series forecasting has become an unusually diverse architectural testbed. Recent progress spans decomposition-based Transformers (Wu et al., 2021; Zhou et al., 2022), simple linear forecasters (Zeng et al., 2023), patch-based Transformers (Nie et al., 2023), temporal 2D modeling (Wu et al., 2023), inverted attention (Liu et al., 2024), and MLP-based mixing architectures (Chen et al., 2023; Wang et al., 2024). These models encode markedly different assumptions about how temporal, cross-variate, periodic, and multiscale structure should be represented. Consequently, forecasting benchmarks do more than measure predictive accuracy: they implicitly compare which architecture/configuration choices are useful for a given dataset and forecast horizon.

Yet the usual benchmark view conflates two different notions of architecture transfer. An architecture may rank highly across many tasks simply because it is *globally strong*; this does not imply that the task-specific preference among architectures is itself transferable. This distinction matters whenever benchmark results are reused across datasets, horizons, or incomplete evaluation matrices. It also differs from the classical algorithm-selection question of choosing the best algorithm for an instance (Rice, 1976): before asking which architecture should be selected, we ask what part of an observed architecture ranking is actually specific to the task. This distinction also has a practical consequence: when a benchmark cannot evaluate every model on every related configuration, observations from nearby horizons of the same dataset may provide a better reuse signal than either an exchangeable completion rule or a globally strong architecture ranking.

We formalize this distinction by decomposing the log pooled risk of architecture m on forecasting task t as

$$L_{m,t} = \mu + \alpha_m + \beta_t + \Gamma_{m,t}, \tag{1}$$

where α_m captures architecture-wide strength, β_t captures task-wide difficulty, and $\Gamma_{m,t}$ is their task-specific interaction. The vector

$$\mathbf{c}_t = \left(-\Gamma_{1,t}, \ldots, -\Gamma_{M,t}\right)^\top \tag{2}$$

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.

defines an *architecture-preference profile*: rather than asking which architectures have low absolute loss, it describes which architectures are unusually well or poorly matched to a task after global effects are removed. We refer to the collection of these profiles across tasks as the *compatibility landscape*.

Performance transfer does not imply preference transfer. Within the primary panel, compatibility agreement is 0.800 within datasets and -0.014 across datasets. Agreement also decreases with log-horizon separation, while two singular directions retain 77.8% of interaction variance and 81.0% of pairwise preferences. The external replication panel reproduces this locality pattern. We therefore distinguish completion for a known dataset, which can reuse related horizons, from zero-shot transfer to an unseen dataset.

Controlled completion compares exchangeable factors, dataset sharing, nearest-horizon reuse, and observable descriptors. On entirely hidden horizon blocks, dataset-structured factors recover 83.4% of pairwise preferences and nearest-horizon reuse reaches 90.8%, versus 43.4% for exchangeable matrix factorization. Matched random groups identify real dataset membership as the transferable coordinate. In contrast, strict leave-one-dataset-out prediction from observable descriptors recovers about 0.63–0.67 of pairwise preferences. Frozen masks then relate evaluation budgets to selection regret separately from preference recovery.

Our contributions follow a phenomenon–attribution–prediction–operation sequence:

- **Preference as a distinct transfer object.** We separate globally reusable architecture strength from residual task-specific preference.
- **Dataset-local preference geometry.** We quantify within-dataset locality, horizon-distance decay, low-dimensional structure, and reproduce the same qualitative geometry on a separately executed panel, alongside seed and candidate-set stability.
- **Attribution and prediction of reusable coordinates.** Matched controls distinguish real dataset membership from arbitrary sharing, while block completion and strict LODO bound transfer beyond known datasets.
- **Operational budget consequence.** Frozen-mask replay relates evaluation budget to raw-risk selection regret; no local rule dominates at every budget.

2 Related Work

Architectural diversity and benchmarking in time-series forecasting. Modern forecasting models encode substantially different inductive biases. Decomposition and frequency-domain modeling motivate Autoformer and FEDformer (Wu et al., 2021; Zhou et al., 2022); DLinear demonstrates that simple linear mappings can remain highly competitive on standard long-horizon benchmarks (Zeng et al., 2023); PatchTST introduces patch-level temporal representations (Nie et al., 2023); TimesNet models temporal variation through a two-dimensional representation (Wu et al., 2023); iTransformer inverts the conventional tokenization of multivariate series (Liu et al., 2024); and TSMixer and TimeMixer explore MLP-based and multiscale mixing architectures (Chen et al., 2023; Wang et al., 2024). Large benchmark efforts have consequently become important for separating architectural improvements from evaluation artifacts. The Monash Forecasting Archive provides heterogeneous forecasting datasets and systematic baseline evaluation (Godahewa et al., 2021), while TSLib evaluates a broad collection of modern time-series architectures under a unified framework (Wang et al., 2026). The latter finds that different architectural structures are suited to different analysis tasks, reinforcing that model effectiveness is task dependent.

These studies compare architectures through their observed predictive performance. Our question is complementary. An architecture can perform well across many tasks because it is globally strong, even if its *relative task-specific advantage* transfers poorly. We therefore separate architecture-wide strength and task-wide difficulty from their interaction and study the resulting architecture-preference profiles directly. This turns task-dependent performance from an endpoint of benchmark comparison into an object whose cross-task structure can itself be analyzed.

Forecasting model selection and meta-learning. Automatically choosing a forecasting method from properties of a target series has a considerably longer history than recent neural architecture

search. It is an instance of the classical algorithm-selection problem (Rice, 1976), in which characteristics of a problem are used to predict the relative suitability of candidate algorithms. Early forecasting work applied meta-learning to select or rank forecasting models from time-series characteristics (Prudêncio & Ludermir, 2004), and subsequent studies expanded this idea to richer feature representations, model ranking, and forecast combination (Lemke & Gabrys, 2010). Feature-based representations have since become a standard way to summarize time-series properties (Lubba et al., 2019), and work on automated forecasting has studied which such characteristics are informative for predicting algorithm performance (Petelin et al., 2023).

A closely related line learns directly from historical forecasting performance. FFORMA uses time-series features to learn forecast-combination weights and obtained strong performance in the M4 competition (Montero-Manso et al., 2020). More recent systems use meta-learning to avoid exhaustive evaluation of candidate forecasting models on a new dataset (Abdallah et al., 2025). Neural architecture and hyperparameter selection has likewise been formulated as transfer from previously evaluated time-series datasets (Moeini et al., 2026). Automatic architecture design has also been studied directly through multi-objective search over composite forecasting architectures (Cao et al., 2026).

Our objective differs from this selection literature in two respects. First, these methods ultimately optimize a *decision*: predicting a strong model, ranking candidate models, weighting forecasts, or searching for an architecture. We instead study the latent relative-preference structure that generates architecture–task interactions after global performance effects are removed. Second, task descriptors or learned task representations are typically used as predictors of model performance or rank. We ask a prior structural question: *how transferable is task-specific architecture preference itself, and along which task coordinates does it vary?* This distinction is important in our experiments: raw performance is substantially more predictable from global architecture effects than the residual compatibility profiles, while the latter exhibit their own dataset–horizon geometry.

Task representations and performance landscapes. Meta-learning more broadly relies on representations that relate properties of datasets to algorithm behavior. Hand-designed meta-features, learned dataset representations, and performance-based representations have all been used to transfer information between learning problems (Rice, 1976; Prudêncio & Ludermir, 2004; Cunha et al., 2016). In time series, compact feature libraries such as CATCH22 summarize properties including autocorrelation, distributions, fluctuation structure, and nonlinear dynamics (Lubba et al., 2019); related forecasting studies use such descriptors to predict algorithm performance (Petelin et al., 2023). These approaches motivate asking whether observable task characteristics provide coordinates for architecture behavior.

Our compatibility landscape is instead defined *from relative architecture behavior*: each task is represented by its interaction profile across architectures after global model and task effects are removed. Two tasks are therefore close when they induce similar architecture preferences, rather than merely when their input statistics or absolute forecasting errors are similar. Empirically, these profiles are strongly aligned across horizons of the same dataset, decay systematically with horizon separation, and occupy a low-dimensional subspace. We use these observations to characterize an *architecture-preference geometry*, rather than assuming a task representation in advance.

Low-rank performance matrices and incomplete evaluation. The reconstruction component of our work is related to collaborative filtering and matrix factorization, where partially observed interaction matrices are represented through low-dimensional latent factors (Koren et al., 2009). Similar ideas naturally arise in algorithm selection: a dataset–algorithm performance table can be viewed as a partially observed interaction matrix, and meta-learning can exploit historical algorithm performance to guide predictions on new problems (Cunha et al., 2016; 2018). This perspective provides a strong baseline for incomplete architecture benchmarking: architectures and forecasting tasks become the two entities whose compatibility interactions must be reconstructed.

Low rank alone, however, treats latent task factors as exchangeable. Our empirical results indicate additional structure: compatibility is highly local to dataset identity while varying systematically with forecast horizon. We therefore use a family of structured factorizations to test whether dataset identity, forecast horizon, and local task variation provide predictive coordinates for compatibility. This distinction becomes particularly important in our block-completion setting. Unlike ordinary

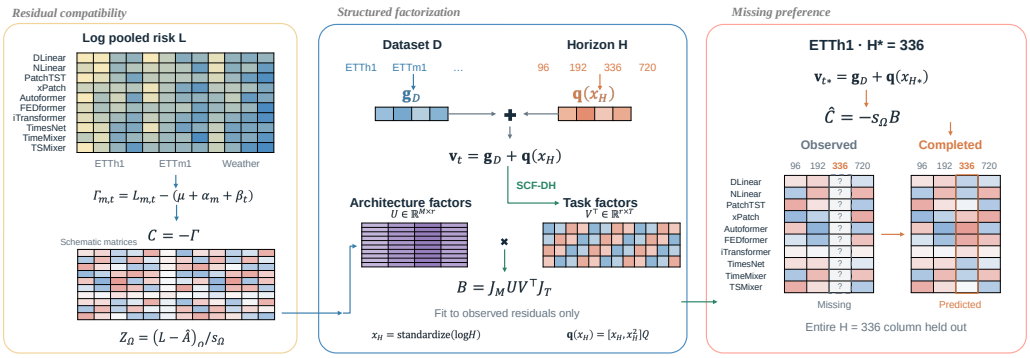

Figure 1: **Compatibility and structured completion.** Residualize log risk for additive architecture and task effects, model task structure with a bilinear factorization, and reconstruct an entirely hidden horizon profile.

entry-wise matrix completion, *all* architecture outcomes for a target $(D, H^\star)$ are hidden. Generic factorization therefore receives no direct observation of the target task, whereas SCF can infer its structured component from the dataset and horizon coordinates shared with observed tasks.

Positioning. Forecasting benchmarks compare observed performance, meta-learning recommends models for new tasks, and collaborative filtering reconstructs incomplete performance matrices. We study a different intermediate object: *task-specific architecture preference after globally transferable performance effects are removed*. The structured factorization family provides controlled predictive probes for dataset identity, horizon, and local task reuse.

3 METHOD

We test which task information predicts residual architecture preference. Figure 1 separates the transfer object, structural hypotheses, and held-out tests. SCF is the controlled factorization family; direct reuse and matched random groups provide complementary controls.

3.1 ARCHITECTURE COMPATIBILITY

Let $R_{m,t,s} > 0$ be the recorded forecasting MSE for architecture m, task $t = (D, H)$, and seed s. We average risks before taking logarithms: $L_{m,t} = \log(\frac{1}{S}\sum_s R_{m,t,s})$. For descriptive analysis of the complete panel, define

$$L_{m,t} = \mu + \alpha_m + \beta_t + \Gamma_{m,t}, \qquad C_{m,t} = -\Gamma_{m,t}, \tag{3}$$

where $\mu = \bar{L}_{\cdot,\cdot}$, $\alpha_m = \bar{L}_{m,\cdot} - \mu$, and $\beta_t = \bar{L}_{\cdot,t} - \mu$. Thus Γ is double-centered, and positive compatibility indicates lower risk than the additive expectation. This residualization is the left stage of Figure 1; it separates task-specific preference from globally reusable architecture strength. The task profile $\mathbf{c}_t = (C_{1,t}, \ldots, C_{M,t})^\top$ describes relative residual preference rather than the risk-minimizing architecture. We use $1 - \rho_S(\mathbf{c}_i, \mathbf{c}_j)$ to describe profile separation.

Interpretation boundary. The panel consists of archived runs. Although the executed update budget and the retained per-run configurations are recorded, the available provenance does not independently verify that every architecture received equally strong hyperparameter tuning (for example, a universal batch size or learning rate). Accordingly, Γ and C should be read as *recorded architecture/configuration compatibility*. Architectural inductive bias is a plausible contributor, but this panel cannot causally separate it from architecture-specific hyperparameter quality or optimization effects. All comparisons below remain within the recorded protocol.

3.2 OPERATIONALIZING STRUCTURAL HYPOTHESES

To operationalize these hypotheses, Structured Compatibility Factorization (SCF) instantiates a family of task representations in a shared bilinear interaction model, $\Gamma_{m,t} \approx \mathbf{u}_m^\top \mathbf{v}_t$ (and hence $C_{m,t} \approx -\mathbf{u}_m^\top \mathbf{v}_t$), with the centering convention specified below. The family isolates alternative predictive coordinates:

$$\begin{aligned} \text{MF}: &\quad \mathbf{v}_t \text{ free}, \\ \text{SCF-D}: &\quad \mathbf{v}_{D,H} = \mathbf{g}_D, \\ \text{SCF-H}: &\quad \mathbf{v}_{D,H} = \mathbf{q}(x_H), \\ \text{SCF-DH}: &\quad \mathbf{v}_{D,H} = \mathbf{g}_D + \mathbf{q}(x_H), \\ \text{SCF-DHR}: &\quad \mathbf{v}_{D,H} = \mathbf{g}_D + \mathbf{q}(x_H) + \boldsymbol{\delta}_{D,H}. \end{aligned} \tag{4}$$

Here x_H is standardized $\log H$, $\mathbf{q}(x_H) = [x_H, x_H^2]\mathbf{Q}$, $\mathbf{g}_D \in \mathbb{R}^r$ is a dataset factor, and $\boldsymbol{\delta}_{D,H} \in \mathbb{R}^r$ is a regularized task residual. MF tests exchangeable low-rank structure; size-matched random groups test arbitrary sharing; SCF-D tests dataset membership; SCF-H and SCF-DH test a shared horizon coordinate alone and conditional on dataset membership. The dataset mean tests local reuse, and nearest-horizon reuse tests whether proximity refines that reuse (Appendix G). The factors are empirical representations of preference, used to compare alternative predictive coordinates.

3.3 OBSERVED-ONLY RESIDUAL FITTING

Let Ω denote the available evaluations. We first estimate additive architecture and task effects by least squares on Ω, $\widehat{A}_{m,t} = \widehat{\alpha}_m + \widehat{\beta}_t$, and freeze them. The remaining observed residuals are scaled by their training-only standard deviation s_Ω. We fit a centered interaction,

$$Z_\Omega = (L - \widehat{A})_\Omega / s_\Omega, \qquad B = J_M U V^\top J_T, \qquad J_n = I_n - \mathbf{1}\mathbf{1}^\top / n, \tag{5}$$

by minimizing

$$\frac{1}{|\Omega|} \sum_{(m,t)\in\Omega} (Z_{m,t} - B_{m,t})^2 + \sum_{\theta \in \Theta_{\text{factors}}} \lambda_\theta \frac{\|\theta\|_F^2}{|\theta|}. \tag{6}$$

The reconstructed compatibility is $\widehat{C} = -s_\Omega B$, while $\widehat{L} = \widehat{A} + s_\Omega B$ supplies a separate risk prediction. Centering uses predicted values, not hidden outcomes. It makes the fitted interaction share the descriptive zero-sum convention, and residual scaling prevents cross-dataset risk magnitudes from determining its numerical scale. The same procedure applies to every component ablation and MF.

Rank and regularization are selected by pairwise recovery on two inner validation splits, with residual MSE breaking ties. Validation targets subtract only additive effects learned from the corresponding inner-training cells. Inner holdouts match the outer regime: entries for random completion, whole horizons for block completion. Appendix F specifies the optimization and validation procedure.

3.4 MISSING-DATA PROTOCOLS AND EVALUATION

The right stage of Figure 1 depicts the stricter entire-horizon protocol. Random completion hides individual architecture–task evaluations while retaining at least three observed architectures per task. Entire-horizon completion excludes all cells of $t^\star = (D, H^\star)$ from fitting and inner validation, while retaining other horizons of D. SCF-D uses $\mathbf{g}_D$; SCF-H and SCF-DH use their horizon-dependent expressions from Equation 4 before predicted interactions are centered. An unobserved free residual must obey

$$\boldsymbol{\delta}_{D,H^\star} = \mathbf{0}. \tag{7}$$

Thus, DHR has no separate block-completion variant. MF has no observation to identify its target task factor.

For a set of eligible architecture pairs $\mathcal{P}_t$, preference accuracy is

$$A_t = \frac{1}{|\mathcal{P}_t|} \sum_{(i,j)\in\mathcal{P}_t} \mathbb{I}\Big[\operatorname{sign}(\widehat{C}_{i,t} - \widehat{C}_{j,t}) = \operatorname{sign}(C_{i,t} - C_{j,t})\Big]. \tag{8}$$

Random completion evaluates pairs whose two entries are hidden; block completion evaluates all 45 pairs. Exact predicted ties agree only with true ties. Profile recovery is $\rho_{\mathrm{S}}(\widehat{\mathbf{c}}_t, \mathbf{c}_t)$. In random completion this uses all ten entries, including observed ones; in block completion all ten are hidden. Selection uses the predicted log-risk ordering and is evaluated separately by $100(R_{\hat{m},t}/\min_m R_{m,t} - 1)$.

All outcome-dependent preprocessing is fitted on the corresponding training cells: inner-training cells for validation and outer-observed cells for refitting. Hidden outcomes define evaluation targets only. Task metadata, including dataset identity and horizon, remain available to every model.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

The complete supervised panel contains ten architectures (DLinear, NLinear, PatchTST, xPatch, Autoformer, FEDformer, iTransformer, TimesNet, TimeMixer, and TSMixer), eleven datasets, and 41 dataset–horizon tasks. Two seeds per cell give 820 completed forecasting runs, using the established 200-update budget. We reuse their recorded outcomes throughout. Appendix A lists the task universe and provenance.

For independent replication of the phenomenon, we use a separately executed, frozen panel of six fixed architectures, ten registered external datasets, and 40 dataset–horizon tasks, with two seeds per cell (480 successful runs). Primary analyses, completion variants, and their hyperparameters were fixed without this panel. A separate update-budget audit compares C1 geometry at 100, 400, and 800 updates with the 200-update reference on 16 tasks from four external datasets. Appendix D gives the protocol and complete audit.

We ask whether preference is local, whether it is low-dimensional, and whether task structure predicts missing preference. The observed-only component study uses 20 shared masks at 40% missingness. Entire-horizon completion exhaustively evaluates 40 blocks across ten multi-horizon datasets, hiding 400 architecture–task cells across folds. PEMS08 has a single task and is excluded from block targets. Pairwise recovery is primary; profile correlation and decision regret are complementary. Dataset-cluster bootstrap preserves all horizons of a sampled dataset. Reported block-effect intervals use 5,000 draws; random-mask variation characterizes sensitivity to missingness within the primary panel.

4.2 PHENOMENON: ARCHITECTURE PREFERENCE IS LOCAL AND REPRODUCIBLE

Dataset and horizon locality. Figure 2 contrasts log-risk with its residual compatibility. Across the complete panel, mean compatibility agreement is 0.800 within datasets versus -0.014 across datasets. This gap limits global reuse of a residual ranking. Within-dataset agreement decreases with absolute log-horizon separation: the slope is -0.215 (95% CI $[-0.318, -0.118]$; permutation $p = 0.0001$). Inference treats the ten multi-horizon datasets, rather than task pairs, as the replication units.

Locality persists under alternative preference definitions. C1 removes architecture-wide strength through double centering. Within-task standardized log-risk (C2) and reverse risk rank (C3) instead retain raw within-task architecture ordering. Both give within-/cross-dataset agreement $0.870/0.439$, with horizon slope -0.114. Removing architecture main effects sharpens locality, while the horizon-distance decline remains. C2 and C3 are rank-equivalent under Spearman correlation.

Independent replication. The separately executed external panel reproduces the central C1 pattern: mean within-dataset profile agreement is 0.722, compared with 0.037 across datasets (gap 0.685; dataset-bootstrap 95% CI $[0.457, 0.888]$), and the within-dataset log-horizon slope is -0.306. Independently constructed C1 profiles agree at 0.931 across its two seeds and preserve 94.0% of pairwise signs. Completion results remain evaluated on the primary panel.

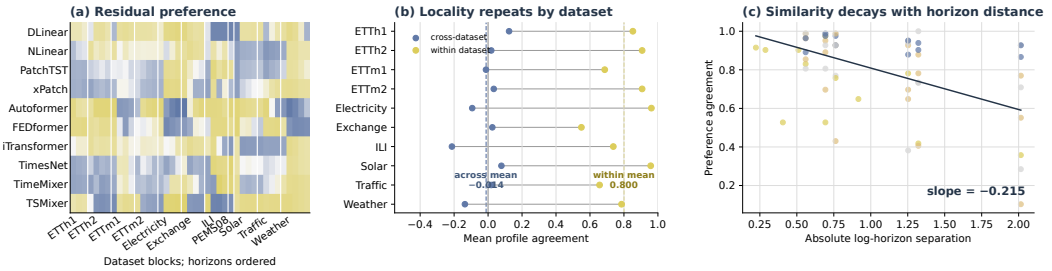

Figure 2: **Dataset-local residual preference.** (a) Centered compatibility matrix. (b) Within- and cross-dataset agreement (means 0.800 and -0.014). (c) Within-dataset agreement versus log-horizon separation (slope -0.215).

Table 1: **Local preference geometry.** Panel A reports primary and external locality with C2/C3 checks. Panel B reports seed, candidate-set, and update-budget stability. Panel C gives descriptive full-panel SVD summaries.

A. Locality and independent replication

Panel / definition	Within ↑	Across	Gap ↑	Slope	Slope 95% CI	p
Primary C1	0.800	-0.014	+0.815	-0.215	$[-0.318, -0.118]$	0.0001
External C1	0.722	0.037	+0.685	-0.306	–	–
Primary C2	0.870	0.439	+0.431	-0.114	$[-0.184, -0.057]$	0.0001
Primary C3	0.870	0.439	+0.431	-0.114	–	–

B. Stability and sensitivity

Check	Profile ρ ↑	Pairwise ↑	Subspace	Slope / scope	Notes
Primary seed C1	0.899	0.905	0.958	$-0.171/-0.259$	2 fixed seeds
External seed C1	0.931	0.940	–	–	2 separate seeds
Primary seed C2	0.925	–	–	–	raw within-task ordering
C1 family exclusion	–	–	0.905–0.999	8/8 negative	8 reduced panels
Update audit vs. 200	0.861–0.921	0.875–0.921	–	100/400/800	4 D; 16 T; 6 A

C. Full-panel low-rank preservation

Rank	Variance (%) ↑	Profile ρ ↑	Pairwise A ↑	Top-1 ↑	Top-3 ↑
1	60.1	0.520	0.714	0.659	0.626
2	<u>77.8</u>	0.743	<u>0.810</u>	0.585	0.732
3	88.5	0.831	0.853	0.610	0.764
4	93.8	0.876	0.880	0.683	0.846

Seed, candidate-set, and update stability. Seed-specific C1 reconstruction yields profile agreement 0.899 (median 0.927; 95% dataset-bootstrap CI $[0.846, 0.946]$), pairwise-sign stability 0.905 (CI $[0.871, 0.936]$), and rank-two subspace cosine 0.958. Both seeds reproduce positive locality gaps and negative horizon slopes. All eight architecture-family exclusions preserve the same pattern; detailed ranges appear in Table 1 and Appendix C.

The geometry persists across tested optimization budgets. The update-budget audit retains profile agreement of 0.861–0.921 and pairwise-sign agreement of 0.875–0.921 relative to 200 updates. Together, the external replication and stability checks reproduce dataset-local preference across the tested panels, seeds, candidate subsets, and update budgets.

Low-dimensional supporting structure. The residual interaction is strongly compressible: rank two explains 77.8% of interaction variance and preserves 81.0% of pairwise preferences. Rank three reaches 88.5% and 85.3%, respectively; rank four reaches 93.8% and 88.0%. Figure 5 visualizes the task trajectories; the full-panel diagnostic is specified in Appendix E.

Table 2: **Attribution of residual preference.** Frozen outer folds and observed-only preprocessing. A is hidden-pair accuracy, ρ profile Spearman, Top-3 risk recall, and R percentage regret. Random groups use twenty size-matched partitions; paired effects and capacity details are in Appendix G.

	Random 40%: 20 masks				Horizon block: 40 folds			
Method	$A\uparrow$	$\rho\uparrow$	Top-3$\uparrow$	$R\%\downarrow$	$A\uparrow$	$\rho\uparrow$	Top-3$\uparrow$	$R\%\downarrow$
Generic MF	0.626	0.572	0.652	9.37	0.434	-0.176	0.675	6.30
Random groups	0.521	0.233	0.639	7.01	0.500	0.017	0.628	8.14
SCF-D	0.786	0.753	0.707	4.60	0.834	0.791	0.742	4.47
SCF-DH	0.769	0.738	0.698	4.56	0.833	0.785	0.742	4.41
Same-dataset mean	0.785	0.734	0.730	6.05	0.882	0.858	0.817	4.35
Nearest horizon	0.786	0.745	0.731	5.74	0.908	0.894	0.867	3.15

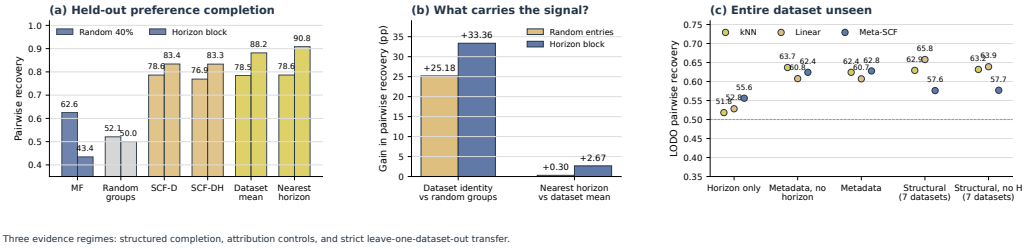

Figure 3: **Preference reconstruction across information regimes.** (a) Random-entry and horizon-block recovery. (b) Dataset and nearest-horizon controls. (c) Strict leave-one-dataset-out recovery from observable descriptors.

4.3 Attribution: Dataset Identity and Horizon Proximity Carry Signal

Table 2 compares exchangeable factors, matched random groups, same-dataset reuse, and structured completion on frozen outer folds. Under 40% random missingness, SCF-D recovers 78.6% of hidden pairwise preferences versus 62.6% for generic MF. Whole-profile Spearman increases from 0.572 to 0.753. Hiding the complete architecture vector of a horizon yields 83.4% accuracy for SCF-D versus 43.4% for MF. Large-margin recovery reaches 88.5% for random entries and 93.1% for blocks (Appendix H). Hidden-pair accuracy is primary.

The gain depends on real dataset structure. Random partitions preserve the number and sizes of the dataset groups and use the identical SCF-D factorization, candidate grid, inner splits, and optimization budget. Their accuracy is 52.1% under random missingness and 50.0% for complete blocks. Relative to these controls, SCF-D improves dataset-average accuracy by 25.18 points (95% CI $[19.44, 29.75]$) and 33.36 points ($[29.47, 37.40]$), respectively. The differences are positive in all eleven random-mask datasets and all ten eligible block datasets (Figure 3b), showing that real membership carries predictive information beyond arbitrary sharing.

Horizon proximity refines dataset-local reuse. With an entire horizon hidden, nearest-horizon reuse recovers 90.8% of pairwise preferences, compared with 88.2% for the same-dataset mean. The paired gain is 2.67 points (95% CI $[1.11, 4.06]$), positive in eight of ten datasets (Figure 3b). Both reuse only outer-observed residuals from other horizons. Under random missingness, the difference is smaller and unresolved: $+0.30$ points ($[-0.76, 1.31]$). The shared horizon polynomial does not improve SCF-D; a free residual gives only an unresolved random-mask increment ($+1.17$ points, 95% CI $[-0.34, 2.65]$). The supported horizon signal is local reuse, not an independent gain from the tested global parameterization. Full family and interior/endpoint analyses are in Appendix G.

4.4 Operational Consequence: Evaluation Budgets and Selection Regret

The preceding analyses establish a scientific and predictive result, but preference recovery need not automatically improve an architecture decision. We replay the frozen matrix masks as evaluation

budgets. Each task reveals $2, 3, 4, 5, 6$, or 8 of ten architecture evaluations; full evaluation supplies the $10/10$ oracle endpoint. Completion methods select the architecture with minimum completed log risk. Pairwise recovery is diagnostic, and selection regret relative to the task oracle is the operational metric.

We include a cross-dataset global-strength baseline to separate broad architecture strength from local preference reuse. For each target dataset, its architecture effects are estimated using observed cells from the other datasets only; the common target-task offset is calibrated from its revealed cells. Figure 4 summarizes the trade-off; full paired-bootstrap results are in Appendix K. Local reuse improves on exchangeable MF most consistently at intermediate budgets ($4/10$–$6/10$). The lowest point regret ranges from 6.75% at $3/10$ to 1.23% at $8/10$. The clearest paired gain over global strength occurs at $6/10$, whereas the $2/10$ condition favors MF (8.89%). The operational pattern is therefore budget-dependent.

4.5 Boundary Analysis: Transfer Weakens on Unseen Datasets

Seen-dataset completion can reuse other horizons of the target dataset, but this information regime is unavailable for a new dataset. We therefore leave one complete dataset out at a time. For each outer fold, architecture-wide log-risk effects are estimated only from the remaining datasets; the held-out dataset contributes neither outcomes to feature preprocessing nor to model or hyperparameter fitting. Its risks are read only after prediction to form an architecture-centred residual compatibility target. Thus, task-wise target centering cannot alter the pairwise metric.

The all-dataset metadata representation contains log horizon, context length, channel count, training-window count, context-to-horizon ratio, sampling interval, and forecast duration in physical time. It excludes dataset identity, all architecture outcomes, and all outcome-derived quantities.

Do observable series properties explain dataset locality? We extract nine outcome-independent statistics from each dataset's chronological training prefix: autocorrelation at lags 1, 12, and 24; normalized spectral entropy; peak spectral energy; linear-trend strength; half-prefix mean and variance shifts; and mean absolute cross-channel correlation. Extraction uses at most 8,192 training rows and 32 deterministically sampled channels per dataset. These are bounded conventional descriptors.

For every inner dataset fold, we re-estimate architecture effects from that inner training set before constructing residual targets; feature imputation, scaling, rank selection, and regression fitting also use training datasets only. This corrects the archived metadata run's reuse of outer-training architecture effects during inner selection. All comparisons use the corrected protocol, the same eleven datasets, and equal dataset weights. The no-horizon panel also removes context-to-horizon ratio.

Table 3 shows that observable metadata transfers only part of preference: pairwise recovery is 60.8–63.0% for the all-dataset metadata panels, compared with 52.2–57.9% for raw-series statistics alone. Combining both reaches 54.1–59.2%. No single parameterization dominates: metadata without horizon fields reaches 63.7% with kNN, while the all-metadata panel reaches 62.8% with Meta-SCF. The tested raw-series statistics do not improve metadata-based prediction, indicating that these conventional statistics do not capture the dataset-local structure observed in the completion experiments.

Horizon is relationally useful; global transfer remains weak. Nearest-horizon reuse improves over the dataset mean on observed datasets. In contrast, horizon-only LODO recovery is 52.8–54.4%, while metadata without any horizon-dependent field reaches 66.7% through kNN. Across all three tested predictors, adding horizon-dependent fields to that reduced metadata does not improve recovery. Together with the SCF-DH control, this supports a dataset-relative interpretation of horizon organization. The held-out dataset contributes no architecture outcomes in LODO, and the best descriptor results (0.63–0.67) remain below relational completion (0.83–0.91). Reuse is strongest within a known dataset; observable coordinates predict only part of preference on a new one.

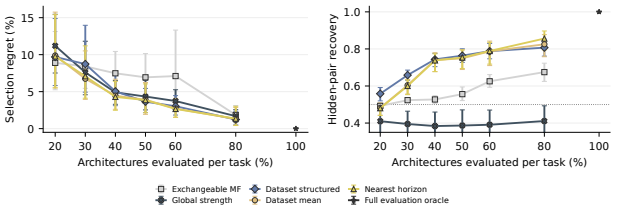

Figure 4: **Evaluation budget versus selection regret.** Frozen masks replay the fraction of architectures evaluated per task. Left: raw-risk regret relative to the ten-model oracle. Right: hidden-pair recovery of residual preference.

Feature set	kNN	Linear	Meta-SCF
Horizon only	0.530	0.528	0.544
Metadata w/o H	0.667	0.629	0.639
Metadata	0.608	0.619	0.630
Series stats.	0.579	0.522	0.522
Metadata + stats.	0.541	0.592	0.555

Table 3: **Strict LODO: observable preference coordinates.** Hidden-pair accuracy averaged over eleven held-out datasets. Inner folds recompute architecture effects and feature preprocessing; target outcomes remain held out.

5 Conclusion

The evidence supports a selective transfer picture: residual preference is reproducible and dataset-local, real dataset identity carries predictive signal, and nearest-horizon reuse helps when a related configuration has been observed. Observable descriptors recover only part of this preference on unseen datasets. The external replication and update-budget audit reproduce the locality pattern across the tested panels and training budgets. Frozen-mask replay links this structure to selection regret at intermediate evaluation budgets. Preference reconstruction and raw-risk model selection remain distinct objectives.

6 Limitations

The evidence concerns bounded architecture and dataset universes under the recorded forecasting protocol. The external replication broadens the empirical panel, but generalization to other architecture families, domains, training objectives, or optimization regimes remains open. The update audit tests only the stated budgets; it neither establishes convergence nor independently verifies equally strong hyperparameter tuning across architectures. Consequently, the residual interaction cannot causally separate architectural bias from architecture-specific configuration quality or optimization effects. Dataset-structured completion and local reuse assume observations at other horizons of the same dataset. Our zero-shot experiment extends prediction to wholly unobserved datasets, but is limited to eleven datasets and nine conventional statistics from capped training prefixes. Transfer from broader raw-series representations and to a new architecture universe remains open.

Reproducibility Statement

We provide the benchmark composition, recorded protocol, source ledger, and random-seed identifiers in Appendix A. The estimation of architecture compatibility, statistical tests, missing-data protocols, SCF optimization, and the reconstruction hyperparameter search grid are specified in Appendices B and F. Additional robustness analyses and dataset-level results are reported in the remaining appendices. We will provide anonymized code and the frozen experiment configurations used to construct the forecasting panel and reproduce all reported figures and tables as supplementary material. An anonymous evaluation release containing the sealed risk ledger, frozen protocol, evaluation scripts, and expected summaries is available at the anonymous reproduction repository. It supports the evaluation-only analyses and does not redistribute raw datasets, checkpoints, or training assets.

AI USE STATEMENT

Generative AI tools were used to assist with language editing, manuscript organization, visualization drafting, code implementation, and experimental analysis. Responsibility for the final manuscript, references, protocol descriptions, and submitted results rests with the authors.

REFERENCES

Mustafa Abdallah, Ryan A. Rossi, Kanak Mahadik, Sungchul Kim, Handong Zhao, and Saurabh Bagchi. Evaluation-free time-series forecasting model selection via meta-learning. *ACM Transactions on Knowledge Discovery from Data*, 2025.

Qianying Cao, Shanqing Liu, Alan John Varghese, Jérôme Darbon, Michael S. Triantafyllou, and George Em Karniadakis. Automatic selection of the best neural architecture for time series forecasting. *Nature Communications*, 17:7058, 2026. doi: 10.1038/s41467-026-73687-9.

Si-An Chen, Chun-Liang Li, Sercan O. Arik, Nathanael Christian Yoder, and Tomas Pfister. TSMixer: An all-mlp architecture for time series forecasting. *Transactions on Machine Learning Research*, 2023.

Tiago Cunha, Carlos Soares, and André C. P. L. F. de Carvalho. Selecting collaborative filtering algorithms using metalearning. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pp. 393–409, 2016. doi: 10.1007/978-3-319-46227-1_25.

Tiago Cunha, Carlos Soares, and André C. P. L. F. de Carvalho. Metalearning and recommender systems: A literature review and empirical study on the algorithm selection problem for collaborative filtering. *Information Sciences*, 423:128–144, 2018. doi: 10.1016/j.ins.2017.09.050.

Rakshitha Godahewa, Christoph Bergmeir, Geoffrey I. Webb, Rob J. Hyndman, and Pablo Montero-Manso. Monash time series forecasting archive. *Neural Information Processing Systems Track on Datasets and Benchmarks*, 1, 2021.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009. doi: 10.1109/MC.2009.263.

Christiane Lemke and Bogdan Gabrys. Meta-learning for time series forecasting and forecast combination. *Neurocomputing*, 73(10–12):2006–2016, 2010. doi: 10.1016/j.neucom.2009.09.020.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations*, 2024.

Carl H. Lubba, Sarab S. Sethi, Philip Knaute, Simon R. Schultz, Ben D. Fulcher, and Nick S. Jones. catch22: Canonical time-series characteristics. *Data Mining and Knowledge Discovery*, 33:1821–1852, 2019. doi: 10.1007/s10618-019-00647-x.

E. Moeini, C. Vox, M. Anastacio, W. Skaf, M. Baratchi, and H. H. Hoos. Neural architecture and hyperparameter selection through meta-learning on time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24405–24413, 2026. doi: 10.1609/aaai.v40i29.39622.

Pablo Montero-Manso, George Athanasopoulos, Rob J. Hyndman, and Thiyanga S. Talagala. FFORMA: Feature-based forecast model averaging. *International Journal of Forecasting*, 36 (1):86–92, 2020. doi: 10.1016/j.ijforecast.2019.02.011.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

Gašper Petelin, Gjorgjina Cenikj, and Tome Eftimov. Towards understanding the importance of time-series features in automated algorithm performance prediction. *Expert Systems with Applications*, 213:119023, 2023. doi: 10.1016/j.eswa.2022.119023.

Ricardo B. C. Prudêncio and Teresa B. Ludermir. Meta-learning approaches to selecting time series models. *Neurocomputing*, 61:121–137, 2004. doi: 10.1016/j.neucom.2004.03.008.

John R. Rice. The algorithm selection problem. In *Advances in Computers*, volume 15, pp. 65–118. Elsevier, 1976. doi: 10.1016/S0065-2458(08)60520-3.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y. Zhang, and Jun Zhou. TimeMixer: Decomposable multiscale mixing for time series forecasting. In *International Conference on Learning Representations*, 2024.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Yong Liu, Chen Wang, Mingsheng Long, and Jianmin Wang. Deep time series models: A comprehensive survey and benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026. doi: 10.1109/TPAMI.2026.3690845.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11121–11128, 2023. doi: 10.1609/aaai.v37i9.26317.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 27268–27286, 2022.

A Benchmark and Implementation Details

Table 4 lists the 41 tasks and retained dataset metadata. Channel counts describe the recorded configuration; unavailable values are marked with dashes and are not filled from external dataset defaults. Ten datasets contribute four horizons each; PEMS08 contributes one. The ten architectures are DLinear, NLinear, PatchTST, xPatch, Autoformer, FEDformer, iTransformer, TimesNet, TimeMixer, and TSMixer. Seeds 20260902 and 20260903 give 820 complete outcomes without imputation.

Table 4: Frozen task universe. Horizons are measured in sampling steps.

Dataset	Frequency	Channels	Context	Horizons	Tasks	Models	Runs
ETTh1	1h	7	96	96, 192, 336, 720	4	10	80
ETTh2	1h	7	96	96, 192, 336, 720	4	10	80
ETTm1	15min	7	96	96, 192, 336, 720	4	10	80
ETTm2	15min	7	96	96, 192, 336, 720	4	10	80
Electricity	1h	321	96	96, 192, 336, 720	4	10	80
Exchange	1d	8	96	96, 192, 336, 720	4	10	80
ILI	1w	–	104	24, 36, 48, 60	4	10	80
PEMS08	5min	170	12	12	1	10	20
Solar	15min	1	96	96, 192, 336, 720	4	10	80
Traffic	1h	862	96	96, 192, 336, 720	4	10	80
Weather	10min	21	96	96, 192, 336, 720	4	10	80

Recorded supervised run configurations record 200 updates and standard normalization. Most tasks use context 96; retained metadata list context 104 for ILI and 12 for PEMS08. The retained DLinear ETTm1 run, for example, uses batch size 256, validation every 20 updates, learning rate 10^{-3}, and chronological train/validation/test lengths $34560/11520/11520$. This example does not independently verify a universal batch size or learning rate across architectures. Derived metadata retain a

2,000-step default and 200-step validation interval; these are not the executed-run budget. We prioritize retained executed configurations and disclose this metadata discrepancy in the artifact audit. We average recorded MSE across seeds before taking logarithms, preserving the original evaluation scale. No cross-dataset raw-MSE average is used as a reconstruction score. The recorded runs used CUDA execution; we make no new common-hardware or runtime claim.

B Compatibility Estimation and Statistics

The descriptive C1 interaction is

$$C^{(1)}_{m,t} = -\left(L_{m,t} - \bar{L}_{m,\cdot} - \bar{L}_{\cdot,t} + \bar{L}_{\cdot,\cdot}\right).$$

It is a finite-panel estimand whose main effects depend on the task and architecture universe. C2 is negative within-task standardized log-risk. C3 is reverse within-task risk rank. Both retain the raw architecture ordering and do not remove architecture main effects; their identical Spearman summaries are therefore expected. C1 within/cross agreement is $0.800/-0.014$, with horizon slope -0.215. Both C2 and C3 give $0.870/0.439$, with slope -0.114. The locality analysis uses 10,000 bootstrap draws and 10,000 horizon permutations. Dataset clusters preserve dependence among horizons; task pairs are not independent replicates. Block effects average within dataset, then resample the ten dataset means 5,000 times. Percentile intervals come from this paired distribution. For random completion, each dataset resample retains all masks and tasks; hidden-pair counts are pooled within each mask before averaging over masks. Point effects use the observed paired statistics. Random-mask standard deviations describe sensitivity to missingness within this panel, not population-level dataset uncertainty.

C Robustness of Preference Locality

Table 5: C2 locality under seed and architecture-family exclusions. Within/Across are mean agreements, Gap their difference, and Slope the log-horizon coefficient. Intervals are dataset-cluster bootstrap CIs.

C2 variant	M	Within	Across	Gap	95% CI	Slope	95% CI
Seed 20260902	10	0.859	0.441	0.418	$[0.364, 0.451]$	-0.101	$[-0.167, -0.043]$
Seed 20260903	10	0.857	0.445	0.412	$[0.336, 0.467]$	-0.113	$[-0.175, -0.060]$
All architectures	10	0.870	0.439	0.431	$[0.368, 0.474]$	-0.114	$[-0.184, -0.057]$
w/o 2d-temporal	9	0.876	0.460	0.416	$[0.347, 0.454]$	-0.110	$[-0.183, -0.052]$
w/o decomposition	9	0.846	0.329	0.517	$[0.443, 0.577]$	-0.136	$[-0.226, -0.062]$
w/o frequency/decomposition	9	0.868	0.476	0.392	$[0.324, 0.448]$	-0.107	$[-0.175, -0.051]$
w/o linear	8	0.867	0.469	0.397	$[0.311, 0.448]$	-0.110	$[-0.188, -0.048]$
w/o mlp-mixing	9	0.869	0.451	0.418	$[0.359, 0.451]$	-0.100	$[-0.186, -0.036]$
w/o multiscale	9	0.878	0.459	0.420	$[0.349, 0.462]$	-0.129	$[-0.215, -0.052]$
w/o patch-based	8	0.846	0.302	0.543	$[0.460, 0.623]$	-0.126	$[-0.195, -0.068]$
w/o variate-centric	9	0.868	0.482	0.386	$[0.320, 0.431]$	-0.115	$[-0.181, -0.058]$

The primary C1 seed audit is reported in Section 4.2; the additional seed and architecture-family controls below use C2. C2 seed-to-seed profile agreement is 0.925 (95% dataset-bootstrap CI $[0.900, 0.949]$). This is stability of within-task risk ordering, complementary to rather than a substitute for the independently centred C1 audit. Across eight declared architecture-family exclusions, C2 agreement remains higher within datasets than across datasets, with negative horizon slopes approximately between -0.136 and -0.100. These controls assess the studied candidate universe and two seeds; they do not establish stability at convergence or for arbitrary new models.

D Independent External Replication and Update-Budget Audit

The external registry, architectures, preprocessing, and task horizons were frozen before execution. The replication panel contains 480 successful runs: six fixed architectures, ten registered external datasets, forty tasks, and two seeds. Primary-panel completion analyses and hyperparameters were fixed before this external panel. Its C1 locality is reported in Table 1: within/across agreement is

$0.722/0.037$, with a locality-gap bootstrap interval of $[0.457, 0.888]$ and horizon slope -0.306. The two seed-specific C1 matrices have mean profile agreement 0.931 and pairwise-sign stability 0.940.

The separately frozen update-budget audit uses 16 tasks from four external datasets and the same six architectures. It measures residual-preference stability at 100, 400, and 800 updates relative to the 200-update reference.

Table 6: External update-budget stability relative to 200 updates. The audit uses 16 tasks, four datasets, and six architectures, and reports preference agreement.

Updates	Profile $\rho \uparrow$	Pairwise sign $\uparrow$	Profile median	Tasks	Datasets	Architectures
100 vs. 200	0.875	0.883	0.886	16	4	6
400 vs. 200	0.921	0.921	0.943	16	4	6
800 vs. 200	0.861	0.875	0.886	16	4	6

D.1 Primary C1 under architecture-family exclusion

For each of the eight previously declared exclusions, we retain all 41 tasks, pool the two seed risks before taking logarithms, and recompute μ, α_m, β_t, and $\Gamma_{m,t}$ on the reduced architecture panel. No full-panel main effect is inherited. Within/across agreement averages task pairs; slope averages the ten within-dataset log-horizon regression slopes. Rank-two energy is descriptive. Subspace cosine averages the two principal cosines between reduced-panel and full-panel right-singular task subspaces; it is rotation-invariant. These overlapping exclusions are sensitivity checks, not eight independent replications or a structured-null test.

Table 7: Recentered C1 sensitivity to architecture-family exclusion. All eight exclusions preserve a positive locality gap and negative horizon slope.

Excluded family	M	Within	Across	Gap	Slope	Rank-2	Cosine
2d-temporal	9	0.799	-0.018	0.817	-0.255	78.5%	0.989
decomposition	9	0.815	-0.003	0.818	-0.216	72.2%	0.905
frequency/decomposition	9	0.796	-0.012	0.808	-0.231	81.6%	0.995
linear	8	0.785	-0.041	0.826	-0.221	81.9%	0.957
mlp-mixing	9	0.781	-0.005	0.786	-0.253	81.5%	0.934
multiscale	9	0.810	-0.011	0.821	-0.206	79.9%	0.980
patch-based	8	0.772	-0.019	0.790	-0.251	78.8%	0.995
variate-centric	9	0.801	-0.009	0.810	-0.218	80.7%	0.999

E Additional Compatibility Geometry

Figure 5 uses the first two right-singular task coordinates scaled by singular values of the complete C1 matrix. All 41 tasks are retained. A consistent four-color palette is reused across datasets, distinguished additionally by marker identity. The trajectories are descriptive, not fitted response functions.

F SCF Implementation and Validation

Optimization. We fit additive effects by minimum-norm least squares on observed cells, then freeze them and standardize the observed residuals by their own SD. All latent products are double-centered using predicted values only. Every model uses float64 Adam, learning rate 0.01, 700 steps, clipping norm 1, and predictions recomputed after the last update. Every method has 36 candidates over ranks $\{1, 2, 3, 4\}$. MF, SCF-D, and SCF-H use base penalties $\{0.0003, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1, 3\}$. SCF-DH uses base penalties $\{0.003, 0.03, 0.3\}$ crossed with horizon penalty multipliers $\{1, 10, 100\}$. SCF-DHR uses, at each rank, the three base penalties with unit multipliers, plus the six fixed (base, horizon multiplier, residual multiplier) triples $(0.003, 1, 10)$, $(0.03, 1, 100)$, $(0.3, 10, 1)$, $(0.003, 10, 100)$, $(0.03, 100, 10)$, and $(0.3, 100, 100)$. Architecture and dataset factors retain the base penalty. This is an equal-trial, bounded search with

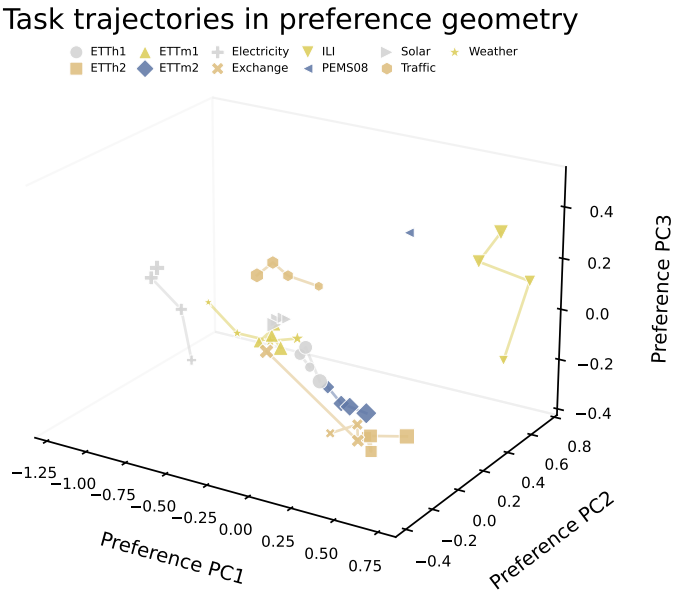

Figure 5: Descriptive SVD trajectories of the complete compatibility panel. Connected points follow increasing horizon within each dataset.

Table 8: Deterministic full-panel low-rank diagnostics. Added and cumulative variance quantify retained interaction energy; these are not held-out results.

Rank	σ_r	Added var. (%)	Cum. (%)	ρ	Pairwise	Top-1	Top-3
1	3.078	60.1	60.1	0.520	0.714	0.659	0.626
2	1.673	17.7	77.8	0.743	0.810	0.585	0.732
3	1.297	10.7	88.5	0.831	0.853	0.610	0.764
4	0.918	5.3	93.8	0.876	0.880	0.683	0.846
5	0.615	2.4	96.2	0.892	0.894	0.878	0.821
6	0.605	2.3	98.5	0.942	0.937	0.902	0.886
7	0.361	0.8	99.3	0.953	0.948	0.927	0.911
8	0.308	0.6	99.9	0.992	0.987	0.951	0.984
9	0.093	0.1	100.0	1.000	1.000	1.000	1.000
10	0.000	0.0	100.0	1.000	1.000	1.000	1.000

family-specific spaces, not a global optimum guarantee. SCF-DHR is evaluated only on random masks. At rank r, their respective latent parameter counts are $51r, 21r, 12r, 23r, 64r$, before the centering constraints; additive fitting is shared. Thus, the comparison does not equate effective capacity.

Each outer fold has two deterministic inner splits. Random splits hold out 30% of the outer-observed entries, requiring at least two inner-training entries per task and architecture. Block splits hide one complete observed horizon per dataset with at least two available horizons. Candidate selection maximizes mean validation pairwise accuracy, with residual MSE breaking ties, followed by rank and penalty. Validation residuals subtract only the inner-training additive fit. Refit preprocessing uses all outer-observed cells and no outer targets. Hidden-value replacement by NaN leaves preparation unchanged; the pilot centering error is below 10^{-12}. The outer target interaction remains the frozen descriptive full-panel estimand, used only for scoring.

Coverage and numerical checks. The study contains 260 outer fits over 20 random masks and 40 horizon blocks, each with two inner splits and 36 candidates. All fits return finite predictions, and all interaction RMS values exceed 10^{-6} on the standardized reference scale. Before validation and final prediction, an interaction with RMS below 10^{-8} on the observed-residual scale is treated as exactly zero, rather than ranking floating-point remnants. The same rule applies to all models. Replacing hidden outcomes with NaN leaves preprocessing and inner-validation scores unchanged. These checks establish implementation invariants, not a guarantee of global optimization convergence.

Table 9: Fold variation for the factorization family. SDs use 20 random masks or 40 horizon blocks; block SD reflects task heterogeneity. Paired dataset-cluster intervals appear in Table 10.

	Random 40%: 20 masks				Horizon block: 40 folds			
Method	SD(A)	SD(ρ)	SD(Top-3)	SD(R)	SD(A)	SD(ρ)	SD(Top-3)	SD(R)
Generic MF	0.057	0.098	0.045	6.21	0.232	0.555	0.192	15.09
SCF-D	0.054	0.072	0.069	2.11	0.091	0.173	0.177	12.22
SCF-H	0.050	0.028	0.024	1.99	0.199	0.492	0.213	14.28
SCF-DH	0.047	0.070	0.067	1.97	0.097	0.190	0.192	12.25
SCF-DHR	0.056	0.092	0.064	1.34	—	—	—	—

Table 10: Paired component effects. Differences are left minus right and ΔA is in percentage points. D/DH/DHR denote SCF-D, SCF-DH, and SCF-DHR; intervals use 5,000 dataset-cluster draws.

Protocol	Contrast	ΔA	CI lo	CI hi	$\Delta\rho$	CI lo	CI hi
Random 40%	D–MF	+16.06	+13.03	+18.74	+0.181	+0.128	+0.237
Random 40%	DH–MF	+14.32	+10.31	+17.68	+0.166	+0.114	+0.216
Random 40%	DH–D	-1.74	-3.38	-0.14	-0.015	-0.036	+0.004
Random 40%	DHR–DH	+0.89	-0.73	+2.50	+0.016	-0.005	+0.039
Block	D–MF	+39.94	+30.56	+49.06	+0.967	+0.737	+1.184
Block	DH–MF	+39.89	+30.11	+49.50	+0.962	+0.726	+1.186
Block	DH–D	-0.06	-1.06	+1.06	-0.005	-0.022	+0.011

G Frozen-Fold Attribution Controls

The attribution controls freeze all backbone outcomes, all outer folds, and the original SCF predictions. Random-group matching replaces the eleven real dataset memberships by twenty seed-fixed random task partitions with exactly the same group-size vector. Each receives SCF-D's same 36-candidate, two-inner-split protocol. Same-dataset mean transports each architecture's observed additive residual over other horizons of its own dataset; nearest horizon transports the residual from the deterministically closest observed log-horizon. Both have no tuned parameters; additive residualization is fitted on observed cells only. A missing source residual falls back to zero. Neither baseline accesses a hidden target value or full-panel statistic.

Table 11: Frozen-fold attribution controls. A is hidden-pair accuracy, ρ profile Spearman, Top-3 risk recall, and R percentage regret. Random groups average twenty matched-capacity partitions.

	Random 40%				Entire horizon block			
Method	$A\uparrow$	$\rho\uparrow$	Top-3↑	$R\%\downarrow$	$A\uparrow$	$\rho\uparrow$	Top-3↑	$R\%\downarrow$
Generic MF	0.626	0.572	0.652	9.37	0.434	-0.176	0.675	6.30
Random-group matched	0.521	0.233	0.639	7.01	0.500	0.017	0.628	8.14
SCF-D	0.786	0.753	0.707	4.60	0.834	0.791	0.742	4.47
SCF-H	0.558	0.166	0.665	6.05	0.542	0.112	0.683	6.26
Same-dataset mean	0.785	0.734	0.730	6.05	0.882	0.858	0.817	4.35
Nearest horizon	0.786	0.745	0.731	5.74	0.908	0.894	0.867	3.15
SCF-DH	0.769	0.738	0.698	4.56	0.833	0.785	0.742	4.41
SCF-DHR	0.778	0.754	0.715	4.35	—	—	—	—

Table 12: Selected latent rank and active task-factor count. Reuse baselines have zero fitted task-side parameters; values are outer-fold means.

	Random 40%		Entire horizon block	
Method	selected r	task-side parameters	selected r	task-side parameters
Generic MF	3.05	125.0	2.45	100.5
Random-group matched	2.46	27.1	2.77	30.4
SCF-D	3.65	40.1	4.00	44.0
SCF-H	2.45	4.9	2.17	4.3
Same-dataset mean	—	0.0	—	0.0
Nearest horizon	—	0.0	—	0.0
SCF-DH	3.20	41.6	4.00	52.0
SCF-DHR	3.45	186.3	—	—

Table 13: Dataset-cluster paired effects for pre-specified attribution contrasts. Positive ΔA favors the left method; intervals use 5,000 dataset-cluster draws.

Protocol	Contrast	ΔA (pp)	95% CI	n_D^+/n_D
Random 40%	SCF-D – Generic MF	+14.09	[+9.11, +18.23]	10/11
Random 40%	SCF-D – Random-group matched	+25.18	[+19.44, +29.75]	11/11
Random 40%	Nearest horizon – Same-dataset mean	+0.30	[-0.76, +1.31]	6/11
Random 40%	SCF-DH – SCF-D	-1.80	[-3.29, -0.47]	2/11
Random 40%	SCF-DHR – SCF-DH	+1.17	[-0.34, +2.65]	8/11
Block	SCF-D – Generic MF	+39.94	[+30.50, +49.33]	10/10
Block	SCF-D – Random-group matched	+33.36	[+29.47, +37.40]	10/10
Block	Nearest horizon – Same-dataset mean	+2.67	[+1.11, +4.06]	8/10
Block	SCF-DH – SCF-D	-0.06	[-1.06, +1.17]	4/10

For the frozen post-hoc horizon partition, SCF-DH minus SCF-D is -0.44 points (95% CI $[-1.89, 1.00]$) for interior blocks and $+0.33$ points (CI $[-1.67, 2.22]$) for endpoints. This analysis refits nothing; it does not select a new horizon model. The controls therefore distinguish the evidence for real dataset identity from descriptive horizon locality, and do not presume that either free residual flexibility or a learned shared factor must dominate direct local reuse.

H Margin-conditioned Preference Recovery

We stratify eligible hidden pairs by the absolute reference margin $\Delta_{ij,t} = |C_{i,t} - C_{j,t}|$. Quartile boundaries are computed once from all 45 pairs of the 41 full-panel C1 profiles, solely for post-hoc evaluation. They are shared across methods and never used for fitting or model selection. We replay the frozen 20 random masks and 40 horizon blocks, verifying every fold's total correct-pair count against its archived result. Accuracy pools eligible hidden pairs within each bin; repeated masks are not independent replicates, and these descriptive curves carry no inferential intervals.

Under random missingness, SCF-D recovery rises from 57.4% in the smallest margin quartile to 88.5% in the largest. For blocks, the corresponding values are 56.3% and 93.1%; nearest-horizon reuse reaches 95.9% in the largest quartile. Block curves are not strictly monotone. Large-margin recovery supports an interpretation beyond counting tiny signs, but absolute margin is neither seed-normalized confidence nor a deployment-risk measure.

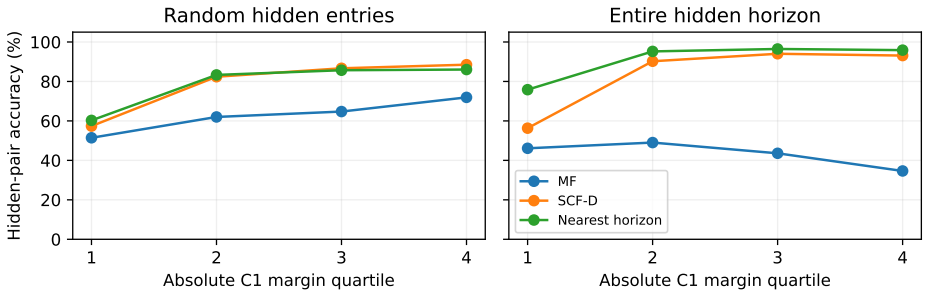

Figure 6: Frozen hidden-pair recovery by evaluation-only absolute C1 margin. Quartiles increase from left to right; methods use identical eligible pairs.

I Dataset and Horizon-Regime Results

Table 14 reports every eligible dataset under the same observed-only protocol. Interior horizons lie strictly between two remaining horizons of their dataset; endpoint horizons lie outside that remaining range. The two regime rows partition the same 40 held-out blocks and require no additional fitting. Neither tests a horizon outside the range observed across all training datasets.

Table 14: Dataset-wise and horizon-regime block completion. D and DH denote SCF-D and SCF-DH. Pairwise gain is relative to MF; R is percentage regret.

		Pairwise accuracy				Profile ρ		Regret (%)	
Dataset	n	MF	D	DH	ΔA	MF	DH	MF	H
ETTh1	4	0.444	0.839	0.844	+40.0	-0.176	0.842	7.39	8.45
ETTh2	4	0.267	0.911	0.906	+63.9	-0.542	0.915	3.11	1.73
ETTm1	4	0.633	0.778	0.767	+13.3	0.355	0.685	0.10	0.10
ETTm2	4	0.383	0.867	0.894	+51.1	-0.309	0.915	0.26	0.08
Electricity	4	0.739	0.944	0.950	+21.1	0.497	0.964	0.31	0.31
Exchange	4	0.389	0.778	0.767	+37.8	-0.255	0.664	9.40	4.17
ILI	4	0.206	0.811	0.844	+63.9	-0.742	0.809	42.10	28.92
Solar	4	0.478	0.928	0.906	+42.8	-0.082	0.927	0.00	0.00
Traffic	4	0.478	0.767	0.756	+27.8	-0.091	0.591	0.00	0.00
Weather	4	0.328	0.717	0.700	+37.2	-0.418	0.539	0.36	0.36
All	40	0.434	0.834	0.833	+39.9	-0.176	0.785	6.30	4.41
Interior	20	0.469	0.862	0.858	+38.9	-0.087	0.844	6.88	3.36
Endpoint	20	0.400	0.806	0.809	+40.9	-0.265	0.726	5.73	5.47

J Missingness and Sparse-Evidence Sensitivity

The additional settings reuse retained masks and the same 36-candidate observed-only estimator. No forecasting backbone is retrained. Each of 20 masks is shared by all five component models. The nominal 20%, 40%, and 60% masks have coverage constraints; their realized missing fractions are reported in the accompanying registry. The 70% masks retain exactly three of ten architectures per task and are identical to the sparse $k = 3$ setting, so they are reused rather than counted as independent evidence.

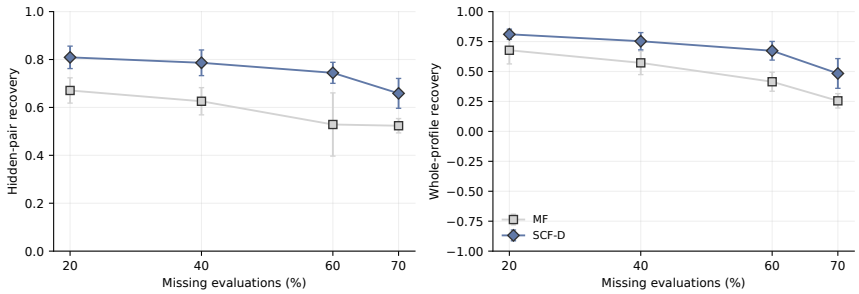

Figure 7: **Incomplete-evidence reconstruction under sparse observations.** SCF-D and generic MF across the frozen missingness masks.

For these additional regimes, inner validation independently masks entries within columns at rate 30%, conditioned on at least two retained values when all columns have more than three observations, and at least one otherwise. For $k = 2$, retaining an observed validation pair requires allowing an empty inner-training column. We therefore require architecture-level coverage and at least one validation pair, but no per-column minimum in that setting. The existing 40% and block inner splits remain fixed. Every split is constructed from outer-observed entries only; target outcomes never select hyperparameters. Thus, the shared estimator is evaluated under explicitly different evidence-availability constraints.

Constant predicted profiles have undefined Spearman correlation. They are not assigned arbitrary ranks or substituted with a favorable score. Profile means and SDs below are conditional on defined values, retaining equal weight per mask with a defined mean. Pairwise accuracy and regret retain all tasks. The registry reports undefined and total profile counts for each method. Four method–setting combinations have 41 undefined profiles out of 820 (one complete mask each): horizon-only at 20%, MF at 60%, dataset-only at $k = 2$, and MF at $k = 5$. All other combinations have none. The realized mean missing fractions at nominal 20%, 40%, and 60% are 20.3%, 39.3%, and 54.5%, respectively; 70% is exact.

Table 15: **Missingness sensitivity.** A is hidden-pair accuracy, ρ profile correlation, Top-3 risk recall, and R percentage regret. SD columns show mask variation.

Missing	Method	A	SD	ρ	SD	Top-3	SD	$R\%$	SD
20%	MF	0.671	0.053	0.677	0.114	0.696	0.029	4.39	0.89
20%	SCF-D	0.809	0.047	0.811	0.043	0.746	0.037	3.02	0.56
20%	SCF-H	0.519	0.136	$0.174^{\dagger}$	0.019	0.678	0.019	5.10	1.31
20%	SCF-DH	0.823	0.037	0.821	0.020	0.752	0.031	3.06	0.54
20%	SCF-DHR	0.819	0.044	0.831	0.018	0.752	0.021	2.98	0.66
40%	MF	0.626	0.057	0.572	0.098	0.652	0.045	9.37	6.21
40%	SCF-D	0.786	0.054	0.753	0.072	0.707	0.069	4.60	2.11
40%	SCF-H	0.558	0.050	0.166	0.028	0.665	0.024	6.05	1.99
40%	SCF-DH	0.769	0.047	0.738	0.070	0.698	0.067	4.56	1.97
40%	SCF-DHR	0.778	0.056	0.754	0.092	0.715	0.064	4.35	1.34
60%	MF	0.529	0.132	$0.414^{\dagger}$	0.079	0.633	0.052	8.64	3.37
60%	SCF-D	0.744	0.044	0.673	0.078	0.669	0.063	5.64	2.06
60%	SCF-H	0.540	0.045	0.151	0.047	0.641	0.044	6.01	2.10
60%	SCF-DH	0.744	0.046	0.673	0.087	0.662	0.055	5.71	2.31
60%	SCF-DHR	0.695	0.068	0.612	0.106	0.656	0.061	6.22	1.92
70%	MF	0.524	0.030	0.255	0.060	0.569	0.064	8.46	3.10
70%	SCF-D	0.659	0.062	0.484	0.124	0.600	0.062	9.24	5.73
70%	SCF-H	0.521	0.024	0.114	0.045	0.593	0.075	8.18	2.77
70%	SCF-DH	0.642	0.062	0.440	0.135	0.603	0.070	8.84	4.45
70%	SCF-DHR	0.608	0.063	0.403	0.115	0.601	0.060	7.38	2.51

Table 16: **Sparse observations: preference versus decision.** Each task has k observed architectures. The $k = 3$ results reuse the 70% masks; all methods use the same observed cells.

k	Method	A	ρ	Top-3	$R\%$	Observed $R\%$	ΔR
2	MF	0.493	0.099	0.551	9.04	12.04	-3.01
2	SCF-D	0.558	0.291	0.543	9.42	12.04	-2.62
2	SCF-H	0.524	0.089	0.557	8.56	12.04	-3.49
2	SCF-DH	0.580	0.258	0.553	11.29	12.04	-0.76
2	SCF-DHR	0.549	0.205	0.542	9.75	12.04	-2.30
3	MF	0.524	0.255	0.569	8.46	7.17	+1.29
3	SCF-D	0.659	0.484	0.600	9.24	7.17	+2.07
3	SCF-H	0.521	0.114	0.593	8.18	7.17	+1.01
3	SCF-DH	0.642	0.440	0.603	8.84	7.17	+1.67
3	SCF-DHR	0.608	0.403	0.601	7.38	7.17	+0.21
5	MF	0.556	0.437	0.630	9.78	3.46	+6.32
5	SCF-D	0.764	0.721	0.685	5.27	3.46	+1.82
5	SCF-H	0.534	0.151	0.664	6.65	3.46	+3.20
5	SCF-DH	0.757	0.709	0.675	4.87	3.46	+1.42
5	SCF-DHR	0.752	0.729	0.689	5.34	3.46	+1.88

K Frozen Evaluation-Budget Audit

The operational curve in Figure 4 re-expresses the archived masks as evaluation budgets. The $2/10$, $3/10$, $4/10$, $5/10$, $6/10$, and $8/10$ conditions respectively reuse the fixed sparse-$k = 2$, random-70%, random-60%, sparse-$k = 5$, random-40%, and random-20% masks (twenty masks each). MF and SCF-D use their frozen observed-only fits. Dataset-mean and nearest-horizon rules are parameter-free and reconstruct each hidden architecture residual solely from outer-observed cells on other horizons of the same dataset; when no source is available, they revert to the observed additive prior. Thus no hidden outcome enters scaling, fitting, candidate selection, or completion. The additional global-strength baseline estimates each architecture's centered mean log-risk effect from observed cells belonging to other datasets, excluding every task from the target dataset. It then estimates a shared target-task offset using only the revealed target cells. Concretely, for target dataset $D^\star$,

$$\begin{aligned}
\hat{\alpha}_m^{(-D^\star)} &= \mathrm{center}_m \,\mathrm{mean}_{s \in \mathcal{S}_m^{(-D^\star)}} \left[L_{m,s} - \mathrm{mean}_{j \in O_s} L_{j,s} \right], \\
\hat{\beta}_t &= \mathrm{mean}_{m \in O_t} (L_{m,t} - \hat{\alpha}_m^{(-D^\star)}), \\
\hat{L}_{m,t} &= \hat{\alpha}_m^{(-D^\star)} + \hat{\beta}_t,
\end{aligned}$$

where $\mathcal{S}_m^{(-D^\star)} = \{s : D_s \neq D^\star,\ m \in O_s\}$. In the regret calculation, the same revealed risks are retained for all methods and only hidden entries use predictions. The complete ten-model panel is used only to score hidden-pair/profile metrics and regret after selection.

Table 17: **Frozen evaluation-budget audit.** Budget denotes architectures evaluated per task. Regret is relative to the ten-model oracle; intervals are dataset-cluster bootstrap CIs. Pairwise recovery scores hidden pairs.

Budget	Method	Regret ↓	95% CI	Pairwise ↑	95% CI	Top-3 ↑
2/10	Exchangeable MF	8.89%	[5.26,13.13]	0.493	[0.481,0.504]	0.551
2/10	Global strength	11.20%	[7.52,15.46]	0.410	[0.334,0.482]	0.520
2/10	Dataset structured	9.69%	[5.82,14.88]	0.558	[0.520,0.592]	0.543
2/10	Dataset mean	9.88%	[5.51,15.75]	0.482	[0.442,0.509]	0.583
2/10	Nearest horizon	9.77%	[5.51,15.41]	0.480	[0.441,0.506]	0.578
3/10	Exchangeable MF	8.41%	[5.44,11.77]	0.524	[0.511,0.535]	0.569
3/10	Global strength	7.60%	[4.58,11.82]	0.395	[0.322,0.464]	0.572
3/10	Dataset structured	8.73%	[4.93,13.97]	0.659	[0.631,0.686]	0.600
3/10	Dataset mean	6.75%	[3.96,10.51]	0.604	[0.556,0.636]	0.609
3/10	Nearest horizon	7.08%	[4.11,11.19]	0.600	[0.553,0.633]	0.606
4/10	Exchangeable MF	7.48%	[4.95,10.44]	0.527	[0.507,0.548]	0.633
4/10	Global strength	4.91%	[3.29,6.50]	0.384	[0.308,0.459]	0.599
4/10	Dataset structured	5.04%	[3.10,7.43]	0.744	[0.704,0.776]	0.669
4/10	Dataset mean	4.45%	[2.62,6.68]	0.740	[0.677,0.780]	0.690
4/10	Nearest horizon	4.29%	[2.48,6.40]	0.738	[0.678,0.774]	0.693
5/10	Exchangeable MF	6.93%	[4.24,10.15]	0.556	[0.521,0.595]	0.630
5/10	Global strength	4.34%	[2.53,6.54]	0.387	[0.301,0.471]	0.624
5/10	Dataset structured	3.64%	[2.17,5.43]	0.764	[0.727,0.801]	0.685
5/10	Dataset mean	3.90%	[1.93,6.24]	0.749	[0.689,0.795]	0.680
5/10	Nearest horizon	3.86%	[2.15,6.05]	0.754	[0.694,0.795]	0.689
6/10	Exchangeable MF	7.11%	[3.30,13.32]	0.625	[0.593,0.662]	0.652
6/10	Global strength	3.73%	[2.35,5.26]	0.391	[0.307,0.470]	0.639
6/10	Dataset structured	3.03%	[1.84,4.46]	0.788	[0.747,0.829]	0.707
6/10	Dataset mean	2.71%	[1.51,4.09]	0.787	[0.710,0.842]	0.730
6/10	Nearest horizon	2.66%	[1.55,3.97]	0.789	[0.718,0.836]	0.731
8/10	Exchangeable MF	1.84%	[0.80,3.12]	0.675	[0.625,0.723]	0.696
8/10	Global strength	1.69%	[0.91,2.57]	0.411	[0.317,0.493]	0.654
8/10	Dataset structured	1.23%	[0.51,2.36]	0.807	[0.761,0.853]	0.746
8/10	Dataset mean	1.31%	[0.45,2.86]	0.825	[0.758,0.877]	0.782
8/10	Nearest horizon	1.35%	[0.42,3.02]	0.856	[0.795,0.897]	0.812
10/10	Full evaluation oracle	0.00%	[0.00,0.00]	1.000	[1.000,1.000]	1.000

The global-strength row provides a decision baseline, not a chance baseline for residual-preference recovery. Its pairwise score compares a global architecture effect with the residual compatibility ordering, which are different estimands. Paired dataset-cluster bootstrap differences in regret are reported in the machine-readable audit (`evidence/evaluation_budget_paired_regret.csv`). Local rules improve over MF most consistently for $4/10$–$6/10$; relative to global strength, a positive paired interval is clearest at $6/10$, so the aggregate point-estimate ordering should not be read as a uniform per-budget advantage.

Retrospective disagreement stratification. We tested whether local regret gain (global-strength regret minus nearest-horizon regret) increases with disagreement between the target task's full architecture ranking and the global ranking estimated without that dataset. Writing $r_t = \mathrm{rank}(L_{\cdot,t})$ and $r_{\mathrm{global}}^{(-D^\star)} = \mathrm{rank}(\hat{\alpha}^{(-D^\star)})$, the diagnostic is $D_t = 1 - \rho_S(r_t, r_{\mathrm{global}}^{(-D^\star)})$. This disagreement uses hidden outcomes only to define a post-hoc stratification variable; it is not a predictor input, selection rule, or tuning target. The task-level analysis is clustered by dataset and resamples all horizons and masks together. The direction is not stable across budgets: the rank association's dataset-bootstrap interval excludes zero only at $6/10$ among the six budgets. High-minus-low gain intervals are positive at $4/10$, $6/10$, and $8/10$, but cross zero at the other budgets. We therefore treat this as a diagnostic, not evidence for a general monotonic moderation law. Table 18 reports every budget without selecting a favorable one.

Table 18: Retrospective association between global-ranking disagreement and nearest-horizon regret gain. Positive gain favors local reuse; intervals use 5,000 dataset-cluster bootstrap draws.

Budget	n_D	Spearman rho (95% CI)	Slope (95% CI)	High–low gain (95% CI)
2/10	11	-0.11 [-0.50,0.36]	-1.12 [-5.40,4.08]	0.11 [-2.18,2.85]
3/10	11	-0.11 [-0.53,0.32]	-2.01 [-7.70,3.88]	-0.82 [-3.51,1.70]
4/10	11	0.37 [-0.01,0.61]	3.01 [-0.21,6.30]	1.96 [0.71,2.91]
5/10	11	0.27 [-0.12,0.55]	2.48 [-0.56,6.16]	1.50 [-0.68,3.08]
6/10	11	0.52 [0.10,0.70]	3.23 [0.25,6.16]	2.31 [1.04,3.16]
8/10	11	0.41 [-0.07,0.68]	1.81 [-2.38,6.14]	1.91 [0.13,3.33]