MODEL REPLACEMENT UNDER LIMITED ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

When a model is retired, unavailable, or too costly to call, one may still want to reproduce its behavior by querying other available models and retaining a small amount of target-specific information that specifies how their responses should be combined. Which available models provide the best replacement when this retained information is limited? We show that the answer can depend fundamentally on the budget. Even when the original model is known completely, two sets of available models can achieve exactly the same replacement error with unrestricted adaptation yet reverse their optimal ordering when the retained state increases from one to two bits, with arbitrary encoders and nonlinear predictors allowed. Replacement error separates into an irreducible component and the error of compressing the target behavior predictable from training observations and the available models' responses. Under Gaussian predictable behavior, the covariance spectrum of this behavior determines the entire information–error tradeoff and gives a necessary and sufficient condition for one model set to dominate another at every budget. Experiments with language models on three multiple-choice tasks show that matching retained-state budgets changes both adapter comparisons and which models are selected, while compact states can preserve much of the attainable performance: on Gemma/HellaSwag, 848 target-specific bits achieve test total variation 0.1951, compared with 0.1933 for full-precision kernel ridge regression.

Code: https://anonymous.4open.science/r/review-artifact-50C6

1 INTRODUCTION

Suppose a target model will no longer be called, for instance after a hosted version is updated and its behavior changes (Chen et al., 2024), but its responses on a particular task are still needed. One way to approximate those responses is to query other models, which we call peers, and train an adapter on paired peer and target responses. At deployment, the adapter combines the peers' responses using parameters learned from the target. When the same peers serve many replaced targets, the state retained for each target is the resource that grows with the number of replacements. This raises a basic selection problem: which peers provide the best replacement when the retained adaptation is limited?

Different peer sets may require different amounts of target-specific state to reach the same replacement error. Prediction-based comparisons already face related questions in probing and representation evaluation, where an adapter's accuracy depends on both its inputs and what it learns (Hewitt & Liang, 2019; Voita & Titov, 2020; Whitney et al., 2020). Here we ask what happens after learning is complete: if the target is fully known, must the same peers remain optimal as the retained adaptation budget changes?

Our first result shows that the optimal choice can change. Two peer sets can expose the same number of independent behavioral components, explain the same total target variation, and attain exactly the same unrestricted replacement error. Nevertheless, their optimal ordering reverses between one and two bits of retained adaptation. The encoder knows the complete target, and optimization ranges over all encoders and nonlinear predictors. The reversal therefore reflects the structure of the available behavior: one set makes a particularly valuable component accessible with one bit, while another makes a more valuable pair accessible with two.

We characterize this structure through the target behavior predictable from training observations and peer responses. A finite adapter state chooses among finitely many prediction functions. Optimal

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

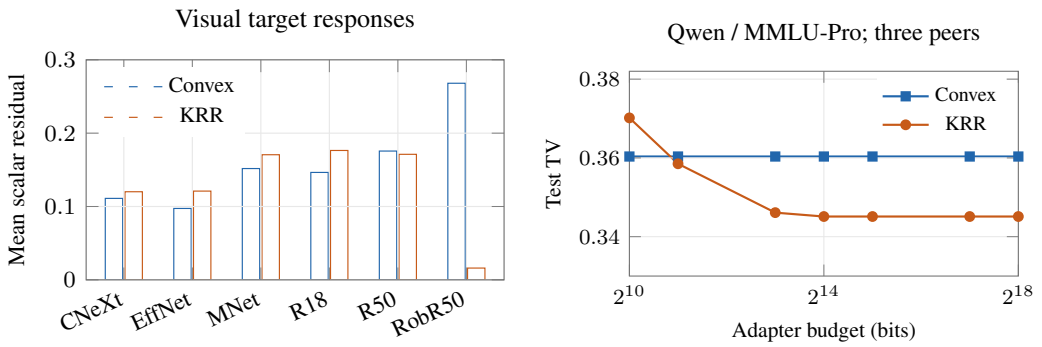

Figure 1: **The choice of adaptation changes replacement comparisons.** Left: in an exploratory scalar-response study, the robust visual target has the largest residual under convex mixing and the smallest under kernel ridge regression (KRR). Right: on Qwen/MMLU-Pro, convex mixing has lower test total variation (TV) at 1,024 bits, whereas KRR has lower TV at 16,384 bits; both use three peers. The visual-study design is given in section C.4; language-model adapters are validation-selected.

replacement is therefore a quantization problem in a space of behaviors, with an additional irreducible error. For Gaussian predictable behavior, the covariance spectrum gives an explicit error curve. Its largest eigenvalue determines the initial benefit of adaptation, its trace determines eventual accuracy, and its dimension and geometric mean determine the approach to that limit. A necessary and sufficient spectral condition then identifies when one peer set is better at every information budget. These results explain both changing and stable replacement orderings.

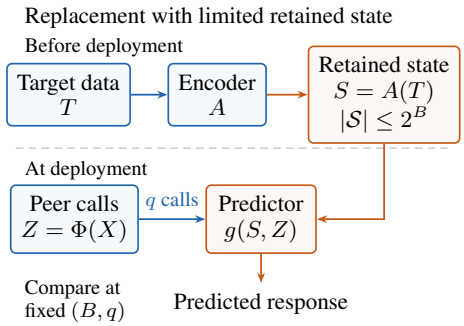

Figure 2: **Replacement with limited retained state.** Target observations become a B-bit state S before deployment. The predictor then combines S with query-dependent peer responses Z. Comparisons fix the state budget B and peer-call count q.

Fitting uses target observations; deployment retains only the target-specific state S. On each new query, the predictor combines S with responses from the chosen peers, without requiring further calls to the target model. The budgets constrain different stages: B bounds retained target information, while q bounds peer calls per query. We compare validation-selected adapters at matched state budgets and peer-call counts on three targets and three multiple-choice tasks. With three calls and 16,384 bits, KRR attains mean test TV 0.2788, compared with 0.2988 for convex mixing. Their ranking reverses between 1,024 and 16,384 bits on Qwen/MMLU-Pro. With KRR and two calls fixed, the validation-selected peer pair also changes with the budget. Compact target-specific states retain much of the full-precision fit on some tasks, and input controls and adapter exchanges show that predictions use both query-aligned peer responses and target-specific adaptation.

These results distinguish *behavioral coverage*, the target behavior that peers eventually expose, from *adaptation efficiency*, how quickly that behavior becomes usable as retained state grows. Peers should therefore be compared at the adaptation budget intended for deployment.

2 Model Replacement with Limited Adaptation

Let $f_\Theta(X)$ be the scalar response of a target drawn from a family indexed by Θ. Training provides observations T about that target. On a fresh query X, a chosen set of peers supplies responses $Z = \Phi(X)$; we call Φ the *peer interface*. The query is independent of (Θ, T), and $\mathbb{E} f_\Theta(X)^2 < \infty$. The peer models, evaluation distribution, and coding rules are fixed independently of the target and treated as public information.

An encoder maps the training observations to a retained state $S = A(T)$. The deployed predictor returns $g(S, Z)$ on each query. With at most B bits of fixed-length state, the smallest achievable mean-squared error is

$$D_\Phi(B) = \inf_{A,g:\,|\mathcal{S}|\le 2^B} \mathbb{E}\big[f_\Theta(X) - g(A(T), \Phi(X))\big]^2. \tag{1}$$

For each fixed interface, the encoder and decoder are fixed across the target family. The state accounts for all target-specific coefficients and decoding choices. When experiments select a peer subset, its identifier is also charged to the state. The budget limits what persists across queries; training computation and target observations may be unrestricted.

We compare interfaces with the same number of peer calls. The predictor observes Z; any additional input features must be included in the interface. Since an interface containing another at no extra cost can reproduce its predictions, our reversal compares nonnested interfaces. Squared error permits an exact characterization of their optimal replacement risks. The experiments study the same selection question for answer distributions, using TV to measure agreement with the target.

3 Optimal Replacement Rankings Can Reverse

We isolate retained-state limitations by setting $T = \Theta$: the encoder knows the complete target, but the deployed predictor receives only its retained state and the chosen peer responses.

Theorem 1 (Equal unrestricted error, reversed finite-state ordering)**.** *There is a bounded target family and two three-response peer interfaces Φ_A, Φ_B such that the exact optimal risks are*

B	0	1	2	≥ 3
$D_A(B)$	20	14	12	10
$D_B(B)$	20	15	11	10.

(2)

The encoder observes the complete target. These values hold both for uniform average target risk and for minimax target risk, allowing arbitrary encoders and nonlinear predictors.

Construction and proof sketch. Let X_i, Θ_i, $i = 1, \dots, 6$, be independent uniform signs. Set

$$f_\theta(X) = \sqrt{6}\theta_1 X_1 + \sqrt{2}\theta_2 X_2 + \sqrt{2}\theta_3 X_3 + \sqrt{5}\theta_4 X_4 + 2\theta_5 X_5 + \theta_6 X_6, \tag{3}$$

with $T = \Theta$, $\Phi_A(X) = (X_1, X_2, X_3)$ and $\Phi_B(X) = (X_4, X_5, X_6)$. The visible components contribute energies $(6, 2, 2)$ and $(5, 4, 1)$. Each set exposes energy ten, while the unobserved query components contribute irreducible error ten.

For visible energies $\nu_1 \ge \nu_2 \ge \nu_3$, the conditional mean is optimal even among nonlinear predictors. If S is nonconstant and binary, let U be its centered, unit-variance version. Then $\mathbb{E}[\Theta_i \mid S] = \mathbb{E}[\Theta_i U]U$. Bessel's inequality gives $\sum_i (\mathbb{E}[\Theta_i U])^2 \le 1$, so one bit can explain at most ν_1 energy. Retaining the largest direction's sign attains the bound.

Two bits quantize the eight coefficient vectors $(\pm\sqrt{\nu_1}, \pm\sqrt{\nu_2}, \pm\sqrt{\nu_3})$ into at most four centroids. Their minimum squared pairwise distance is $4\nu_3$. For a cluster C of size m, its sum of squared distances to the centroid is $\frac{1}{m}\sum_{u<v\in C}\|u - v\|^2 \ge 2(m-1)\nu_3$. Summing over at most four clusters and dividing by eight gives error at least ν_3. Retaining the largest two signs attains it. Three bits retain all visible signs. Each attaining rule has the same query risk for every target, which also proves minimax optimality. The full proof, including randomization and rescaling to $[0, 1]$, is in section A.1.

At one bit, A exposes a component of energy six, compared with B's five. At two bits, B's largest pair has energy nine, compared with A's eight. This allocation explains the reversal despite equal total predictable energy and unrestricted error. More generally, equal-sum ordered triples with $\nu_1^A > \nu_1^B$ and $\nu_3^A > \nu_3^B$ give the same one-versus-two-bit reversal.

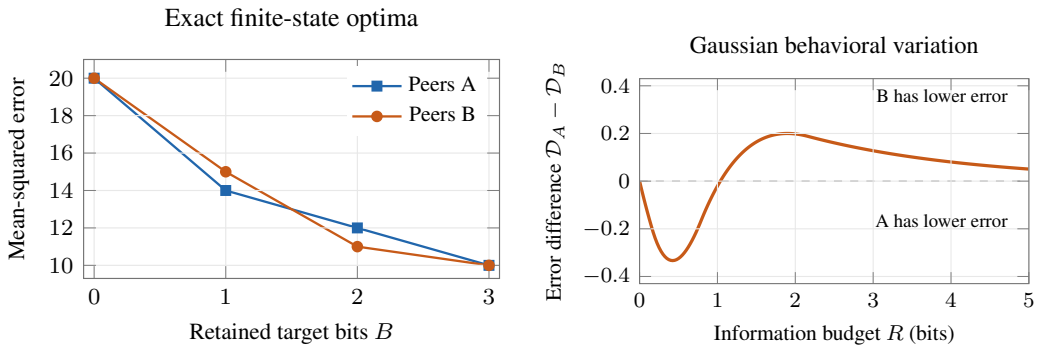

Figure 3: **Equal unrestricted error, different optimal choices.** Left: the exact fixed-bit risks in theorem 1; A is better at one bit and B at two. Right: the Gaussian information-budget model in theorem 3, with behavioral spectra $(6, 2, 2)$ and $(5, 4, 1)$. Negative error difference favors A. Both constructions have unrestricted error ten; they use fixed-cardinality and mutual-information constraints, respectively.

4 What Determines the Value of Adaptation?

4.1 The Behavior That Must Be Represented

Let $h_T(z) = \mathbb{E}[f_\Theta(X) \mid T, Z = z]$ be the best target prediction using the training observations and peer responses. As the target and its training observations vary, h_T is a random element of $\mathcal{H} = L^2(P_Z)$. Its unrestricted error is $D_\Phi^{\text{full}} = \mathbb{E}(f_\Theta(X) - h_T(Z))^2$.

Lemma 2 (Behavioral quantization)**.** *For every nonnegative integer B,*

$$D_\Phi(B) = D_\Phi^{\text{full}} + \inf_{c_1,\dots,c_{2^B} \in \mathcal{H}} \mathbb{E} \min_{j \le 2^B} \|h_T - c_j\|_{\mathcal{H}}^2. \tag{4}$$

Conditional-expectation orthogonality separates irreducible error from approximation of h_T. A B-bit state selects at most 2^B prediction functions, with nearest-function assignment minimizing error for a fixed codebook, as in classical quantizer design (Lloyd, 1982). The object being encoded is predictable behavior; section A.7 gives an example in which preserving this behavior requires fewer bits than preserving the fitted state.

4.2 A Spectral Characterization

The quantization error depends on the full distribution of h_T. Suppose $\mathbb{E}_\Theta f_\Theta(x) = 0$ and h_T is a centered finite-rank Gaussian random function, with nonzero covariance eigenvalues $\nu_1 \ge \cdots \ge \nu_r > 0$ and total target energy $V = \mathbb{E} f_\Theta(X)^2$. Each eigenvalue measures variation along an orthogonal direction of predictable behavior. We now replace the cardinality constraint by $I(T; S) \le R$ and denote the optimal risk by $\mathcal{D}_\Phi(R)$. This relaxation permits randomized, continuous states.

Theorem 3 (Spectral value of retained adaptation)**.** *Under these assumptions,*

$$\mathcal{D}_\Phi(R) = V - \max_{\substack{r_i \ge 0 \\ \sum_i r_i \le R}} \sum_{i=1}^{r} \nu_i (1 - 2^{-2r_i}) \tag{5}$$

$$= V - \sum_i \nu_i + \sum_i \min(\nu_i, \eta), \qquad R = \frac{1}{2} \sum_i \log_2^+(\nu_i/\eta). \tag{6}$$

At $R = 0$, any $\eta \ge \nu_1$ gives the stated risk.

Proof sketch. The orthogonal eigen-coordinates of h_T are independent Gaussians. A coordinate of variance ν_i reconstructed with mean-squared error $d_i < \nu_i$ requires at least $\frac{1}{2} \log_2(\nu_i/d_i)$ bits of mutual information, and a Gaussian test channel attains this value. Allocating the information budget gives $d_i = \min(\nu_i, \eta)$. Composing the channel with the deterministic map $T \mapsto h_T$ yields a feasible

state; data processing gives the converse. Applying the Gaussian rate–distortion solution (Cover & Thomas, 2006) to h_T gives the stated error; section A.4 supplies the full argument.

The spectrum distinguishes initial adaptation benefit, eventual error, and the rate of approach to that error:

$$-\mathcal{D}'_\Phi(0^+) = 2\ln 2\,\nu_1, \qquad \mathcal{D}_\Phi(\infty) = V - \sum_i \nu_i, \qquad \mathcal{D}_\Phi(R) - \mathcal{D}_\Phi(\infty) = r\Big(\prod_i \nu_i\Big)^{1/r} 2^{-2R/r}, \tag{7}$$

where the last equality holds once all directions are active. The strongest predictable direction governs initial benefit; total energy governs the limit; dimension and geometric mean govern the fully active tail. Equal-rank, equal-trace spectra with $\nu_1^A > \nu_1^B$ and $\prod_i \nu_i^A > \prod_i \nu_i^B$ therefore reverse order. The spectra in theorem 1 satisfy these conditions (products 24 and 20), giving the distinct Gaussian illustration in figure 3.

Since a B-bit state obeys $I(T;S) \le B$, we have $\mathcal{D}_\Phi(B) \le D_\Phi(B)$. Ordering these lower bounds does not by itself order fixed-state risks; theorem 1 proves the finite-state reversal directly. The covariance of h_T reflects both training information and deployment-time peer responses, as the linear model in section A.6 illustrates.

4.3 When a budget-independent ordering exists

In the Gaussian information-budget model, a stable ordering can be expressed by pricing retained information instead of fixing its amount. The following transform is the largest prediction gain after charging $2\ln 2\,\tau$ per information bit:

$$\Psi_\nu(\tau) = \sum_{\nu_i > \tau} \left[\nu_i - \tau - \tau\ln(\nu_i/\tau)\right], \qquad \tau > 0. \tag{8}$$

Theorem 4 (All-budget comparison)**.** *For two Gaussian predictable-behavior interfaces to the same centered target problem,*

$$\mathcal{D}_A(R) \le \mathcal{D}_B(R) \text{ for every } R \ge 0 \quad \Longleftrightarrow \quad \Psi_{\nu^A}(\tau) \ge \Psi_{\nu^B}(\tau) \text{ for every } \tau > 0. \tag{9}$$

Proof sketch. Set $G_\nu(R) = V - \mathcal{D}_\Phi(R)$. Pricing information at $2\ln 2\,\tau$ per bit and optimizing each coordinate gives $\Psi_\nu(\tau) = \sup_{R\ge 0}\{G_\nu(R) - 2\ln 2\,\tau R\}$. Closed concave conjugate duality (Rockafellar, 1970) gives the inverse representation $G_\nu(R) = \inf_{\tau>0}\{\Psi_\nu(\tau) + 2\ln 2\,\tau R\}$. Each operation preserves pointwise order, proving both directions.

All-budget dominance can thus be checked from the spectra alone, without choosing an operating budget.

5 Experiments

5.1 Experimental setup

The primary targets are Qwen3.5-9B, Gemma-4-12B-it, and Granite-3.3-8B-Instruct; reference peers are Qwen3-8B, Olmo-3-7B-Instruct, and Ministral-3-8B-Instruct. We normalize first-position answer-code logits over the 10, 4, or 2 options in MMLU-Pro, HellaSwag, or RewardBench (Wang et al., 2024; Zellers et al., 2019; Lambert et al., 2024). RewardBench is a two-response preference task. All adapters, including convex mixing, operate in orthonormal logit-contrast coordinates (section B.1). Agreement is mean $\mathrm{TV}(p,\hat{p}) = \frac{1}{2}\|p - \hat{p}\|_1$.

Fit/validation/test sizes are 2,048/512/1,024 for MMLU-Pro and HellaSwag, and 1,024/256/512 for RewardBench, with disjoint text-duplicate clusters. We compare convex mixing, linear ridge regression, random Fourier features (RFF), Gaussian-kernel ridge regression (KRR), and a one-hidden-layer MLP. For each budget, validation TV selects the peer subset, regularization, rank or width, and encoding before test evaluation. Section B specifies the partitions, candidate grids, and held-out-domain evaluation.

Table 1: **Adapter comparisons at matched state budgets.** Test TV for all nine primary target–task combinations, with three peer calls. Bold marks the best among adapters shown within each column group. Convex results at 16,384 bits equal the 1,024-bit column and are excluded from the right-group bolding. The final row averages the nine combinations.

		1,024-bit cap		16,384-bit cap			
Target	Task	Convex	KRR	Linear	RFF	KRR	MLP
Qwen	MMLU-Pro	**0.3604**	0.3702	0.3536	0.3507	**0.3451**	0.3531
	HellaSwag	0.2004	**0.1931**	0.2026	**0.1912**	0.1927	0.1989
	RewardBench	0.1470	**0.1422**	0.1446	0.1424	**0.1422**	0.1557
Gemma	MMLU-Pro	0.5190	**0.5128**	0.4814	0.4797	**0.4719**	0.4964
	HellaSwag	0.2775	**0.1951**	0.2010	0.1942	0.1937	**0.1916**
	RewardBench	0.1594	**0.1426**	0.1569	0.1517	0.1431	**0.1427**
Granite	MMLU-Pro	**0.5189**	0.5324	0.5255	0.5289	**0.5216**	0.5292
	HellaSwag	0.2851	**0.2758**	0.2929	0.2787	0.2750	**0.2712**
	RewardBench	**0.2214**	0.2236	0.2226	**0.2222**	0.2236	0.2226
Mean		0.2988	0.2875	0.2868	0.2822	0.2788	0.2846

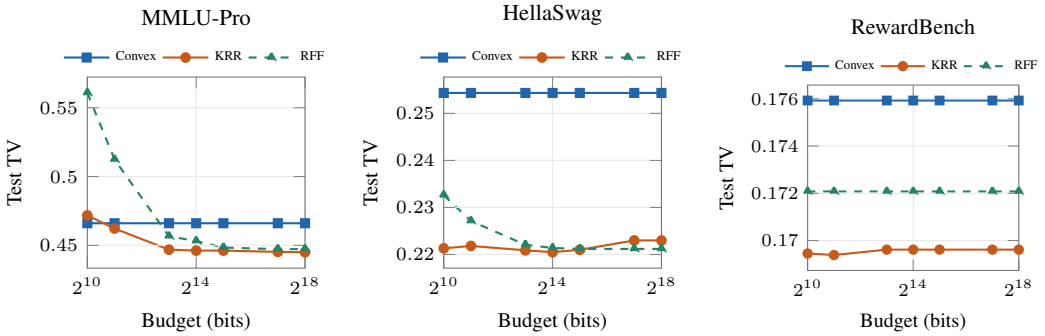

Figure 4: **Additional adaptation has different value across tasks.** Mean test TV over the three primary targets, with three peer calls. Each point uses the adapter selected on validation data within the indicated state cap. Lines connect evaluated budgets. Section C.2 gives all target–task curves, including linear and MLP adapters.

We vary peer calls q per query and target-specific state bits, including coefficients, decoding metadata, and a 176-bit header. Peer-only fitting responses, bases, and deterministic transforms are shared across targets and excluded from this incremental budget (section B.4). We call the original fitted numerical representation *full precision*. Means weight the target–task combinations equally.

5.2 Matching budgets changes the preferred adapter

At full precision, KRR has lower error than convex mixing in eight of nine target–task combinations (mean TV 0.2784 versus 0.3028; table 6), but the two adapters use very different state sizes. Table 1 instead fixes both the state cap and the number of peer calls.

KRR has lower test TV than convex mixing in six of nine combinations at 1,024 bits and seven at 16,384 bits. On Qwen/MMLU-Pro, KRR-minus-convex TV changes from $+0.0098$ (paired 95% interval $[0.0014, 0.0179]$) to -0.0152 ($[-0.0205, -0.0098]$) (section B.7). On Granite/MMLU-Pro and Granite/RewardBench, convex mixing remains better than all four more expressive adapters at 16,384 bits. Thus the empirical advantage of a more flexible adapter depends on the target, task, and available state.

The task means in figure 4 show that KRR changes little beyond the smallest displayed HellaSwag budget, whereas MMLU-Pro benefits more from additional state. On RewardBench, whose response has a single contrast coordinate, some compact adapters outperform the full-precision fit; because validation selects regularization and representation jointly, test error need not decrease monotonically with the cap.

Table 2: **Different state budgets favor different peer pairs.** Validation-selected KRR adapters for Qwen/MMLU-Pro, with two peer calls. Target Qwen3.5 and peer Qwen3 are distinct models. State includes coefficients and decoding metadata.

Cap (bits)	Selected peers	Representation	State (bits)	Validation TV	Test TV
1,024	Olmo + Ministral	rank 16, 3-bit	896	0.3728	0.3708
2,048	Qwen3 + Ministral	rank 32, 4-bit	1,616	0.3584	0.3504
8,192	Qwen3 + Ministral	rank 32, 12-bit	3,920	0.3482	0.3427

Table 3: **Additional peer responses improve replacement at fixed state.** Mean test TV across the nine primary target–task combinations, under a 16,384-bit cap. For one or two calls, validation selects the peer subset. All methods in a row use the same call count.

Peer calls	Linear	RFF	KRR	MLP
One	0.3109	0.3109	0.3062	0.3102
Two	0.2902	0.2858	0.2839	0.2872
Three	0.2868	0.2822	0.2788	0.2846

5.3 The adaptation budget changes peer selection

The validation-selected peer set can change with the budget even when the adapter family and call count are fixed. For KRR on Qwen/MMLU-Pro with two calls, selection chooses Olmo plus Ministral at 1,024 bits and Qwen3 plus Ministral at the displayed 2,048- and 8,192-bit caps (table 2). Under an 8,192-bit cap, the selected state uses 3,920 bits and attains test TV 0.3427, compared with 0.3708 for the 896-bit state selected under the smallest cap. Peers and adaptation size are therefore coupled choices in practice, as they are at the optimum in theorem 1.

Increasing the number of peers. At a 16,384-bit cap, increasing q from one to two to three lowers mean KRR TV from 0.3062 to 0.2839 to 0.2788 (table 3). All four displayed families improve on average, with the larger gain from the second call. Additional responses and additional retained state therefore provide distinct ways to improve replacement.

5.4 Compact adaptation preserves much of the prediction benefit

A small adapter can retain most of the performance of a much larger fitted state. On Gemma/HellaSwag, KRR selected under 1,024 bits attains test TV 0.1951, compared with 0.1933 for the full-precision KRR fit and 0.2775 for budget-matched convex mixing. Its 848-bit state stores rank-16, 12-bit coefficients in a peer-derived basis shared across targets, plus a 176-bit header and per-column scales (section B.4).

Across nine combinations, KRR selected under 1,024 bits increases test TV by 0.0091 on average relative to the FP64 fit: 0.0275 on MMLU-Pro, 0.0019 on HellaSwag, and -0.0020 on RewardBench. Its target-specific states are 67–1,317 times smaller (table 7). A three-coefficient nonnegative calibration of the peers already recovers 55.6% of the full-precision convex-to-KRR improvement (section C.6), indicating that much of the benefit comes from simple rescaling of peer responses.

5.5 Peers and target-specific adaptation both contribute

The compact results raise a mechanistic question: does the adapter use the peers' responses, or would similarly sized uninformative inputs suffice? We refit and select adapters under the same 16,384-bit cap after replacing their inputs. Shuffled rows, random addresses, and constant inputs raise mean KRR TV from 0.2788 to 0.6090, 0.6067, and 0.6033; RFF shows the same pattern (figure 5). A 256-feature input-text TF-IDF control gives KRR TV 0.6073 and RFF TV 0.6066. Query-aligned peer responses therefore carry predictive information that neither misaligned peer features nor a simple text representation provides.

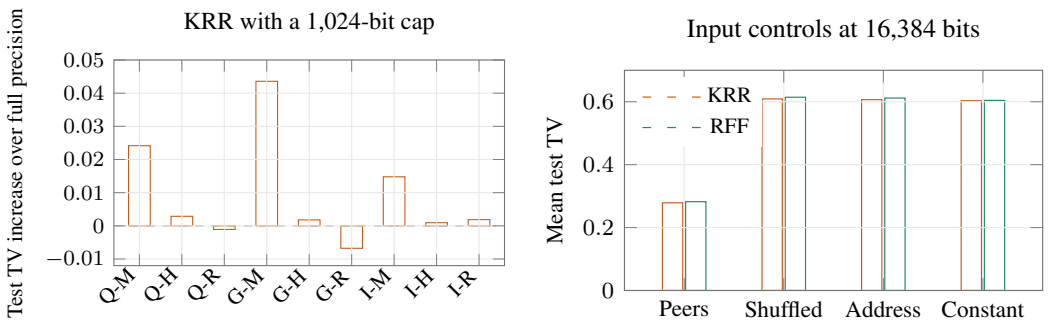

Figure 5: **Compact adaptation uses predictive structure in peer responses.** Left: change in test TV from full-precision KRR to the KRR selected under 1,024 bits; positive values indicate increased disagreement. Q/G/I denote Qwen/Gemma/Granite and M/H/R denote MMLU-Pro/HellaSwag/RewardBench. Right: mean test TV after refitting KRR and RFF on each input condition at 16,384 bits. Shuffling rows, using random addresses, or using constant inputs substantially increases error.

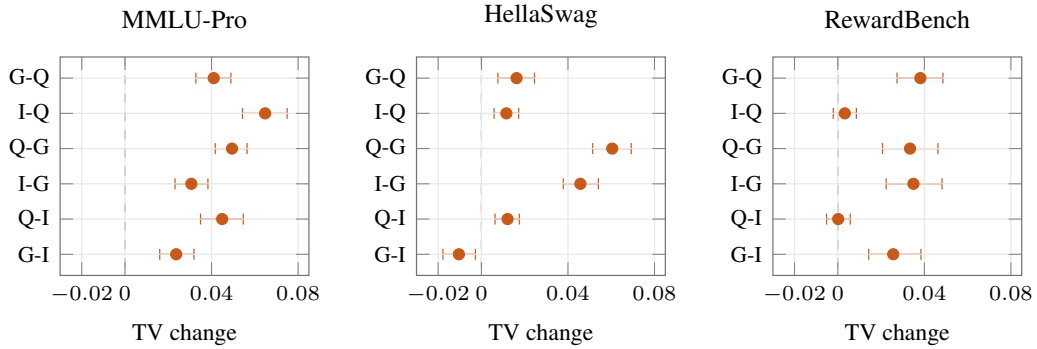

Figure 6: **Adapters retain useful target-specific information.** Change in test TV when a full-precision KRR adapter is replaced by another target's adapter for the same task and peers. Labels indicate donor–recipient targets (Q: Qwen; G: Gemma; I: Granite). Positive values favor the recipient's own adapter. Bars are paired 95% bootstrap intervals over held-out query units.

Holding task and peer responses fixed, we exchange full-precision adapters between targets. Another target's KRR state increases test TV in 17 of 18 ordered exchanges, with 15 pointwise paired 95% intervals above zero (figure 6). The mean increase is 0.0292, compared with 0.0278 for RFF and 0.0266 for MLP. Together, the interventions show that peers supply query-dependent structure and the fitted adapter specializes its use to the target.

5.6 Budget sensitivity extends across peer and target choices

We next vary the models assigned to each role. An alternative peer set (Mistral-7B-Instruct-v0.3, Falcon3-7B-Instruct, and Qwen2.5-7B-Instruct) also serves as secondary targets, predicted from either the reference peers or the primary targets. This gives four groups of nine comparisons, each excluding the target from its own peer set (figure 7).

At full precision, KRR has lower test TV than convex mixing in 8/9, 8/9, 8/9, and 6/9 cells across these groups. At 16,384 bits, the counts are 7/9, 5/9, 8/9, and 6/9, with mean differences -0.0200, -0.0049, -0.0227, and -0.0114. KRR retains an average advantage in all four groups, but budget matching reduces its favorable comparisons from eight to five for primary targets with alternative peers. Adaptation efficiency thus depends on which peers supply the responses.

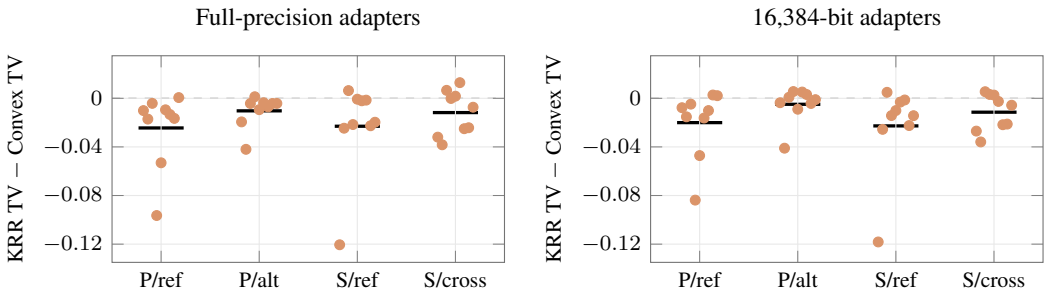

Figure 7: **Changing peers and targets preserves budget-sensitive comparisons.** KRR-minus-convex test TV at full precision (left) and 16,384 bits (right). Points are target–task comparisons; marks show nine-comparison means. Negative values favor KRR. P/S denote primary/secondary targets; ref/alt/cross denote reference peers, alternative peers, or primary targets as peers.

6 Related Work

Model reuse and adaptation. Distillation transfers teacher or ensemble behavior to a student (Hinton et al., 2015); fine-tuning on proprietary outputs tends to reproduce style more readily than capability (Gudibande et al., 2024). Stacking combines predictors with nonnegative least-squares weights (Breiman, 1996); our convex and nonnegative calibration adapters follow this idea. Parameter-efficient adaptation limits task-specific changes to an existing model (Hu et al., 2022), while fine-tuning can succeed in a very low-dimensional subspace (Aghajanyan et al., 2021). Routing balances quality and cost among callable models (Ong et al., 2024). Here retained adaptation combines callable peer responses; we study how limiting its state changes the best replacement.

Prediction-based model comparison. Probe controls separate information in a representation from what a probe learns (Hewitt & Liang, 2019). MDL probing measures description length (Voita & Titov, 2020), and learning-curve evaluation studies rankings across training sample sizes (Whitney et al., 2020). Transferability estimators rank pretrained models by how well a target task can be predicted from their features (You et al., 2021). Usable information depends on the predictor family (Xu et al., 2020), while model stitching can obscure differences in information content (Smith et al., 2025). Our construction instead fixes complete target knowledge and equal unrestricted error, then varies finite retained state over optimal predictors.

Rate–distortion and kernel approximation. Rate–distortion theory studies model compression (Gao et al., 2019) and Bayesian excess risk (Hafez-Kolahi et al., 2021). We use the classical Gaussian solution (Cover & Thomas, 2006) after reducing replacement to compression of predictable behavior. Low-rank kernel approximation (Bach, 2013) and random Fourier features (Rahimi & Recht, 2007) supply the empirical adapter families.

7 Discussion and Conclusion

Equal unrestricted replacement error can coexist with opposite optimal choices at finite budgets. Behavioral quantization identifies what adaptation must represent; the Gaussian characterization explains its budget dependence and determines when the optimal ordering is stable.

Scope. The theory fixes the query distribution and observed inputs, and the complete spectral characterization assumes Gaussian predictable behavior. The experiments evaluate feasible, validation-selected adapters on answer distributions rather than global optima, and do not estimate the behavioral covariance. State bits measure incremental adaptation given shared context, and peer calls count input resources rather than total deployment cost. Extending the analysis to open-ended generation is left to future work.

The practical implication is that peers and retained adaptation should be chosen together: two peer sets with the same eventual accuracy can differ in how much of that accuracy is reachable at a given budget, so comparisons should be made at the budget intended for deployment.

Reproducibility Statement

Complete proofs are in section A; section B specifies models, partitions, adapters, encodings, selection, and uncertainty estimates. An anonymous implementation and review artifacts are available at https://anonymous.4open.science/r/review-artifact-50C6. The repository provides frozen protocol and state records, revision pins, aggregate result tables, and instructions for rebuilding the reported figures and tables. Executable checks verify the finite construction and Gaussian identities. The mirror does not include the complete per-item target and peer logit cache needed to refit adapters and repeat validation selection; this step requires recollecting responses from the specified checkpoints, datasets, and collection environment.

Ethics Statement

The study uses public benchmark tasks and involves no human participants. Reproducing a model's answer distribution can reproduce its errors and biases. Similar average responses can also conceal differences on rare or safety-critical inputs, so deployment decisions require evaluation of the intended use and failure conditions. The adaptation budget measures retained target-specific state, without a privacy or deletion guarantee. Model and dataset use is subject to their respective terms.

AI Use Statement

Generative AI tools, including ChatGPT and coding assistants, assisted with conceptual development, mathematical claims and proofs, analysis of experimental results, design of post hoc analyses, code generation and checking, literature identification, and manuscript and LaTeX-figure preparation. The finite constructions and analytical quantities were also checked with the accompanying executable scripts. This assistance extended beyond language editing. The authors are responsible for the claims, proofs, references, and experimental results.

References

Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 7319–7328, 2021. doi: 10.18653/v1/2021.acl-long.568.

Francis Bach. Sharp analysis of low-rank kernel matrix approximations. In *Proceedings of COLT*, volume 30 of *Proceedings of Machine Learning Research*, pp. 185–209, 2013. URL https://proceedings.mlr.press/v30/Bach13.html.

Leo Breiman. Stacked regressions. *Machine Learning*, 24(1):49–64, 1996. doi: 10.1007/BF00117832.

Lingjiao Chen, Matei Zaharia, and James Zou. How is ChatGPT's behavior changing over time? *Harvard Data Science Review*, 6(2), 2024. doi: 10.1162/99608f92.5317da47.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2 edition, 2006.

Weihao Gao, Yu-Han Liu, Chong Wang, and Sewoong Oh. Rate distortion for model compression: From theory to practice. In *Proceedings of ICML*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2102–2111, 2019. URL https://proceedings.mlr.press/v97/gao19c.html.

Arnav Gudibande, Eric Wallace, Charlie Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. The false promise of imitating proprietary language models. In *International Conference on Learning Representations*, 2024.

Hassan Hafez-Kolahi, Behrad Moniri, Shohreh Kasaei, and Mahdieh Soleymani Baghshah. Rate-distortion analysis of minimum excess risk in Bayesian learning. In *Proceedings of ICML*, volume 139 of *Proceedings of Machine Learning Research*, pp. 3998–4007, 2021. URL https://proceedings.mlr.press/v139/hafez-kolahi21a.html.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of EMNLP-IJCNLP*, 2019. URL https://aclanthology.org/D19-1275/.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. URL https://arxiv.org/abs/1503.02531.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2106.09685.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, L. J. Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. RewardBench: Evaluating reward models for language modeling. *arXiv preprint arXiv:2403.13787*, 2024. URL https://arxiv.org/abs/2403.13787.

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

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs with preference data. *arXiv preprint arXiv:2406.18665*, 2024. URL https://arxiv.org/abs/2406.18665.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, volume 20, 2007.

R. Tyrrell Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, NJ, 1970.

Damian Smith, Harvey Mannering, and Antonia Marcu. Functional alignment can mislead: Examining model stitching. In *Proceedings of ICML*, volume 267 of *Proceedings of Machine Learning Research*, pp. 55972–55998, 2025. URL https://proceedings.mlr.press/v267/smith25a.html.

Elena Voita and Ivan Titov. Information-theoretic probing with minimum description length. In *Proceedings of EMNLP*, pp. 183–196, 2020. doi: 10.18653/v1/2020.emnlp-main.14. URL https://aclanthology.org/2020.emnlp-main.14/.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. *arXiv preprint arXiv:2406.01574*, 2024. URL https://arxiv.org/abs/2406.01574.

William F. Whitney, Min Jae Song, David Brandfonbrener, Jaan Altosaar, and Kyunghyun Cho. Evaluating representations by the complexity of learning low-loss predictors. *arXiv preprint arXiv:2009.07368*, 2020. URL https://arxiv.org/abs/2009.07368.

Yilun Xu, Shengjia Zhao, Jiaming Song, Russell Stewart, and Stefano Ermon. A theory of usable information under computational constraints. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/2002.10689.

Kaichao You, Yong Liu, Jianmin Wang, and Mingsheng Long. LogME: Practical assessment of pre-trained models for transfer learning. In *Proceedings of ICML*, volume 139 of *Proceedings of Machine Learning Research*, 2021.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of ACL*, 2019. URL https://aclanthology.org/P19-1472/.

A Proofs and Further Theoretical Results

Throughout, the encoder and decoder are measurable and fixed before the target is drawn. The fresh query has a distribution independent of the target and training observations. All target-independent public information is conditioned on. We assume the relevant conditional expectations admit versions in the separable space $L^2(P_Z)$; finite-dimensional constructions automatically satisfy this condition. All predictions use the specified peer interface Z; any additional observable features are incorporated into Z.

A.1 Reversal with equal unrestricted error

We prove theorem 1 for average risk under the uniform target prior and for minimax target risk. Let Θ_i and X_i, $i = 1, \dots, 6$, be mutually independent uniform signs, with f_Θ given by equation (3). For either interface, let J denote its three visible coordinates and let $\nu_1 \geq \nu_2 \geq \nu_3$ denote their squared coefficients. In the two instances, these are $(6, 2, 2)$ and $(5, 4, 1)$.

Reduction to three-dimensional quantization. For any retained state $S = A(\Theta)$, the invisible-query contribution has conditional mean zero, including conditional on Θ and the visible queries. Its squared error contribution is exactly ten. The visible conditional mean, optimal among all measurable predictors, is

$$g^*(S, X_J) = \sum_{i \in J} \sqrt{\nu_i}\, \mathbb{E}[\Theta_i \mid S] X_i. \tag{10}$$

Query orthogonality gives the average risk

$$10 + \sum_{i \in J} \nu_i\, \mathbb{E}(\Theta_i - \mathbb{E}[\Theta_i \mid S])^2 = 20 - \sum_{i \in J} \nu_i\, \mathbb{E}\big(\mathbb{E}[\Theta_i \mid S]\big)^2. \tag{11}$$

With zero bits, the signs' means are zero, giving risk twenty. With three bits, the visible signs can be stored exactly, giving risk ten, which the invisible queries make optimal at every larger budget.

The encoder's knowledge of the invisible target signs does not evade a bound on coding the visible signs. Conditional on a visible coefficient vector, that extra knowledge only randomizes the choice of a codeword. For any fixed codebook, the average squared distance of a randomized choice is at least the distance of a nearest deterministic choice. Off-span components of an arbitrary prediction function only add error, so projection onto the three visible query coordinates is optimal. This reduces the coding problem to quantizing eight equiprobable coefficient vectors, even when the original encoder observes all six target signs.

One-bit converse and achievability. If S is constant, it explains no visible energy. Otherwise, write $p = \Pr(S = 1) \in (0, 1)$ and

$$U = \frac{\mathbf{1}\{S = 1\} - p}{\sqrt{p(1-p)}}. \tag{12}$$

Every centered square-integrable function of a binary S is proportional to U. Consequently,

$$\mathbb{E}[\Theta_i \mid S] = \mathbb{E}[\Theta_i U] U, \qquad \mathbb{E}\big(\mathbb{E}[\Theta_i \mid S]\big)^2 = (\mathbb{E}[\Theta_i U])^2. \tag{13}$$

The independent signs Θ_i form an orthonormal set in L^2. Bessel's inequality therefore yields

$$\sum_{i \in J} (\mathbb{E}[\Theta_i U])^2 \leq \mathbb{E}U^2 = 1, \qquad \sum_{i \in J} \nu_i (\mathbb{E}[\Theta_i U])^2 \leq \nu_1. \tag{14}$$

Storing the sign of the largest visible coefficient attains explained energy ν_1. Hence the optimal one-bit risks are $20 - 6 = 14$ and $20 - 5 = 15$. This bound applies to every binary partition of the target family, and the coordinate-sign encoder attains it.

Two-bit converse and achievability. The eight coefficient vectors are

$$\mathcal{V} = \{(\pm\sqrt{\nu_1}, \pm\sqrt{\nu_2}, \pm\sqrt{\nu_3})\}. \tag{15}$$

Their minimum squared pairwise distance is $\delta^2 = 4\nu_3$. A code with at most four states partitions these vertices into $K \le 4$ nonempty clusters, whose optimal representatives are their centroids. For a cluster C containing m vertices, the variance identity gives

$$\sum_{u \in C} \|u - \bar{u}_C\|^2 = \frac{1}{m} \sum_{\substack{u,v \in C \\ u<v}} \|u - v\|^2 \ge \frac{m-1}{2}\delta^2. \tag{16}$$

Summing and using the uniform distribution,

$$\frac{1}{8} \sum_{C} \sum_{u \in C} \|u - \bar{u}_C\|^2 \ge \frac{8-K}{16}\delta^2 \ge \nu_3. \tag{17}$$

Pair vertices that differ only in the smallest coordinate. The four resulting centroids retain the two largest coefficients and discard the smallest, attaining error ν_3. Adding the invisible error gives $10 + 2 = 12$ for A and $10 + 1 = 11$ for B.

Minimax equality and randomization. For a fixed code, worst-target query risk is at least its average over the uniform target prior, so the Bayes converse lower-bounds the minimax risk. Each attaining coordinate-sign code above has the same query risk for every target. It therefore also attains the minimax lower bound. Randomized decoder outputs cannot improve squared error over their conditional mean. A private randomized encoder cannot improve nearest-codeword assignment for any fixed decoder. Conditioning on any public random seed reduces to a deterministic codebook; its average risk obeys the same lower bound. The minimax lower bound follows again by averaging over the uniform target prior.

Bounded output and non-isolated reversal. The target is bounded by $L = \sqrt{6} + 2\sqrt{2} + \sqrt{5} + 3$ in absolute value. The transformation $\widetilde{f} = (1 + f/L)/2$ places it in $[0, 1]$. A corresponding affine transformation of predictors scales every risk by $1/(4L^2)$ without changing the ordering or state size. More generally, for two positive ordered triples with the same sum t, the construction on disjoint query coordinates has total energy $2t$, one-bit risk $2t - \nu_1$, two-bit risk $t + \nu_3$, and three-bit risk t. Thus the strict inequalities $\nu_1^A > \nu_1^B$ and $\nu_3^A > \nu_3^B$ produce a strict reversal on an open subset of the equal-sum parameter surface.

A.2 Proof of the behavioral quantization identity

Let $h_T(Z) = \mathbb{E}[f_\Theta(X) \mid T, Z]$. Since $S = A(T)$, every deployed prediction $g(S, Z)$ is measurable with respect to (T, Z). Conditional-expectation orthogonality gives

$$\mathbb{E}(f_\Theta(X) - g(S,Z))^2 = \mathbb{E}(f_\Theta(X) - h_T(Z))^2 + \mathbb{E}(h_T(Z) - g(S,Z))^2. \tag{18}$$

The fresh-query independence assumption allows the second term to be written as $\mathbb{E}_T\|h_T - g(A(T), \cdot)\|^2_{L^2(P_Z)}$ using one fixed query measure. A state taking at most $M = 2^B$ values selects at most M prediction functions $c_j = g(j, \cdot)$. For this codebook, every encoder has distortion at least $\mathbb{E}\min_j \|h_T - c_j\|^2_{\mathcal{H}}$. Conversely, for any finite codebook, choosing a nearest codeword (with a fixed measurable tie rule) and deploying it realizes that distortion. Infima over codebooks prove equation (4). Attainment of the global infimum is not required.

The same decomposition holds for a stochastic state generated from T with fresh randomness independent of (Θ, X) conditional on T. One conditions additionally on that randomness. Randomization is unnecessary for the fixed-cardinality optimum but is useful in the information-constrained relaxation.

A.3 A general finite-state spectral bound

The following bound connects the finite-state construction to general predictable behavior without a Gaussian assumption. Suppose $\mathbb{E}_\Theta f_\Theta(x) = 0$ for almost every x. Then h_T is centered as an element of $\mathcal{H}$. Let its trace-class covariance have eigenvalues $\nu_1 \ge \nu_2 \ge \cdots \ge 0$, and let $V = \mathbb{E} f_\Theta(X)^2$.

Proposition 5 (Finite-state explained-energy bound)**.** *For an integer $B \ge 0$,*

$$D_\Phi(B) \ge V - \sum_{i=1}^{2^B - 1} \nu_i, \tag{19}$$

with missing eigenvalues treated as zero. The bound is exact at one bit for the sign construction in theorem 1, but need not be tight for general distributions or larger budgets.

Proof. For a state with $M \le 2^B$ positive-probability values, the optimal reconstructed behavior is $m_S = \mathbb{E}[h_T \mid S]$. Because $\mathbb{E}m_S = 0$, its M values span a subspace L of dimension at most $M-1$. Let P_L be the orthogonal projector. Conditional Jensen gives

$$\mathbb{E}\|m_S\|^2 = \mathbb{E}\|\mathbb{E}[P_L h_T \mid S]\|^2 \le \mathbb{E}\|P_L h_T\|^2 = \operatorname{tr}(P_L \operatorname{Cov}(h_T)) \le \sum_{i=1}^{M-1} \nu_i. \tag{20}$$

The final step is the variational characterization of the sum of the leading covariance eigenvalues. Combining orthogonal projections, $\mathbb{E}(f_\Theta(X) - g^*(S,Z))^2 = V - \mathbb{E}\|m_S\|^2$. The result follows by optimizing over states. At one bit in the sign construction, the leading eigenvalue is the largest visible energy and the coordinate-sign encoder attains it.

For a noncentered target, first subtract the target-independent mean response and its induced public mean prediction. The energy in the bound is the energy of this centered target.

A.4 Proof of the Gaussian spectral characterization

Assume h_T is a centered finite-rank Gaussian element of $\mathcal{H}$, with orthonormal eigenfunctions $e_1, \dots, e_r$:

$$h_T = \sum_{i=1}^{r} \xi_i e_i, \qquad \xi_i \text{ independent}, \quad \xi_i \sim \mathcal{N}(0, \nu_i). \tag{21}$$

Since the target is pointwise centered, orthogonality gives $D_\Phi^{\text{full}} = V - \sum_i \nu_i$.

Consider states drawn from a channel $P_{S|T}$, with $I(T;S) \le R$ in bits. Projection of the decoder onto the span of the e_i cannot increase error. Its best coefficients are the conditional means $\widehat{\xi}_i = \mathbb{E}[\xi_i \mid S]$. Write $d_i = \mathbb{E}(\xi_i - \widehat{\xi}_i)^2$, so $0 \le d_i \le \nu_i$.

Converse. Because ξ is a deterministic function of T, data processing gives $I(\xi;S) \le I(T;S)$. Independence of the ξ_i and subadditivity of conditional differential entropy yield

$$I(T;S) \ge I(\xi;S) = \sum_i h(\xi_i) - h(\xi \mid S) \tag{22}$$

$$\ge \sum_i \{h(\xi_i) - h(\xi_i \mid S)\} \ge \frac{1}{2} \sum_i \log_2(\nu_i/d_i). \tag{23}$$

The last inequality uses the Gaussian maximum-entropy bound and Jensen's inequality for the conditional variance. A zero d_i is interpreted by a limiting argument and requires infinite information when $\nu_i > 0$. Hence every admissible reconstruction has total distortion at least the value of

$$\min \sum_i d_i \quad \text{subject to} \quad 0 < d_i \le \nu_i, \quad \frac{1}{2} \sum_i \log_2(\nu_i/d_i) \le R. \tag{24}$$

Achievability. For each desired $0 < d_i \le \nu_i$, use a joint Gaussian pair satisfying the backward representation $\xi_i = \widehat{\xi}_i + \varepsilon_i$, where $\widehat{\xi}_i \sim \mathcal{N}(0, \nu_i - d_i)$ and $\varepsilon_i \sim \mathcal{N}(0, d_i)$ are independent. Generate $S = (\widehat{\xi}_i)_i$ from ξ through the corresponding conditional channel. Then $I(\xi;S) = \frac{1}{2}\sum_i \log_2(\nu_i/d_i)$ and the squared reconstruction error is $\sum_i d_i$. As the channel depends on T only through ξ, the chain rule gives $I(T;S) = I(\xi;S)$. Therefore the converse is attained under the information constraint.

Optimal allocation. The Karush–Kuhn–Tucker conditions for equation (24) give $d_i = \min(\nu_i, \eta)$. Equivalently, allocate $r_i = \frac{1}{2}\log_2^+(\nu_i/\eta)$ to direction i, so its explained energy is $\nu_i(1 - 2^{-2r_i})$. This proves equations (5) and (6). At zero budget set all $d_i = \nu_i$; as $R \to \infty$, all distortions tend to zero.

Initial benefit, limiting error, and convergence rate. An infinitesimal rate is optimally assigned among directions of maximum variance, with total initial explained-energy slope $2 \ln 2 \, \nu_1$. The limiting error is $V - \sum_i \nu_i$. If $\eta < \nu_r$, all r directions are active and

$$R = \frac{1}{2} \log_2 \frac{\prod_i \nu_i}{\eta^r}, \qquad \eta = \Big(\prod_i \nu_i \Big)^{1/r} 2^{-2R/r}. \tag{25}$$

Summing r distortions proves the last equality in equation (7). An adequate threshold for the fully active regime is $R > \frac{1}{2} \sum_i \log_2(\nu_i/\nu_r)$.

If two spectra have equal trace and rank, $\nu_1^A > \nu_1^B$ gives A strictly smaller error for all sufficiently small positive rates. If also $\prod_i \nu_i^A > \prod_i \nu_i^B$, B has strictly smaller error whenever both spectra are fully active. Continuity ensures a crossing between these ranges. For $(6, 2, 2)$ and $(5, 4, 1)$, the products are 24 and 20. This proves the Gaussian reversal with equal unrestricted error used in figure 3, independently of the discrete construction.

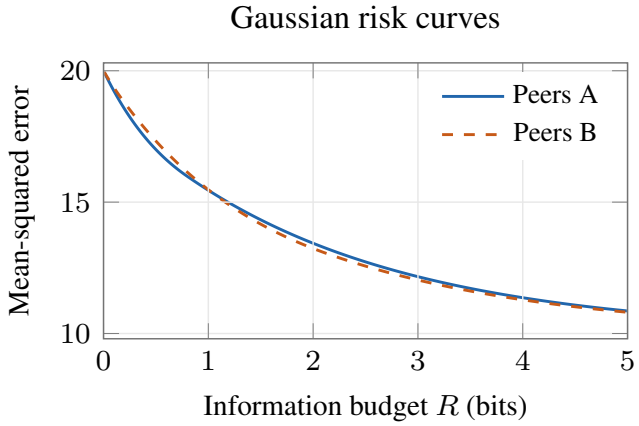

Figure 8: **A common limiting error can conceal a change in optimal ordering.** Gaussian information-budget curves from equation (6) for spectra $(6, 2, 2)$ and $(5, 4, 1)$, with target energy 20. Both approach error ten. The difference plot in figure 3 resolves the crossing.

Relation to fixed-state constraints. For any fixed-length B-bit state, $I(T; S) \le H(S) \le B$, hence $\mathcal{D}_\Phi(B) \le D_\Phi(B)$. The achieving Gaussian state is generally continuous, so this information constraint is a relaxation of literal finite storage. The finite-state result in theorem 1 establishes the reversal separately under a cardinality constraint. For non-Gaussian behavior, the full distribution in equation (4), rather than its covariance alone, determines the quantization error.

A.5 Proof of the all-budget comparison theorem

Define the gain

$$G_\nu(R) = \max_{\substack{r_i \ge 0 \\ \sum_i r_i \le R}} \sum_i \nu_i (1 - 2^{-2r_i}), \qquad R \ge 0. \tag{26}$$

It is continuous, nondecreasing, and concave, with $G_\nu(0) = 0$ and finite upper bound $\sum_i \nu_i$. Price one bit at $a = 2 \ln 2 \, \tau > 0$. Since the least budget realizing an allocation is its sum,

$$\sup_{R \ge 0} \{ G_\nu(R) - aR \} = \sum_i \sup_{r_i \ge 0} \{ \nu_i (1 - 2^{-2r_i}) - a r_i \}. \tag{27}$$

The coordinate derivative vanishes at $2^{-2r_i} = \tau/\nu_i$ when $\nu_i > \tau$; otherwise $r_i = 0$ is optimal. Substituting gives

$$\sup_{R \ge 0} \{ G_\nu(R) - 2 \ln 2 \, \tau R \} = \sum_{\nu_i > \tau} [\nu_i - \tau - \tau \ln(\nu_i/\tau)] = \Psi_\nu(\tau). \tag{28}$$

Closed concavity and conjugate duality (Rockafellar, 1970) give the inverse representation

$$G_\nu(R) = \inf_{\tau>0}\{\Psi_\nu(\tau) + 2\ln 2\,\tau R\}. \tag{29}$$

It can also be checked directly using the water level at $R > 0$. At $R = 0$, take $\tau \geq \nu_1$; limits as $\tau \downarrow 0$ cover the asymptotic endpoint.

If $G_A(R) \geq G_B(R)$ for all R, subtracting the same price and taking the supremum proves $\Psi_A(\tau) \geq \Psi_B(\tau)$ for every $\tau > 0$. Conversely, if the latter inequality holds, adding the same price term and taking the infimum in equation (29) proves $G_A(R) \geq G_B(R)$ for all R. Both interfaces have the same target energy V, so gain ordering is the reverse of error ordering. This proves theorem 4.

Coordinatewise domination of ordered eigenvalues, after zero padding, is a sufficient condition: the same allocation gives at least as much gain in the larger spectrum. The transform condition characterizes the full class of ordered pairs, including those beyond coordinatewise domination. The equal-unrestricted-error examples show why common rank and trace leave this ordering unresolved.

A.6 Training observability and peer visibility in a linear model

A linear-Gaussian model makes the joint role of training information and peer responses explicit. Let

$$f_\Theta(X) = a(X)^\top\Theta, \quad \Theta \sim \mathcal{N}(0,\Sigma), \quad T = L\Theta + \epsilon, \quad \epsilon \sim \mathcal{N}(0,\Omega), \quad \Omega \succ 0, \tag{30}$$

where X, Θ, and ϵ are independent apart from the displayed observation. Define

$$m(z) = \mathbb{E}[a(X) \mid Z = z], \qquad H = \mathbb{E}[m(Z)m(Z)^\top], \tag{31}$$

$$\Sigma_{\text{learn}} = \Sigma L^\top(L\Sigma L^\top + \Omega)^{-1}L\Sigma. \tag{32}$$

Conditional independence yields

$$h_T(z) = m(z)^\top\mathbb{E}[\Theta \mid T], \qquad \operatorname{Cov}(\mathbb{E}[\Theta \mid T]) = \Sigma_{\text{learn}}. \tag{33}$$

The nonzero covariance eigenvalues of h_T are the nonzero eigenvalues of

$$H^{1/2}\Sigma_{\text{learn}}H^{1/2}, \tag{34}$$

by the equality of nonzero spectra of the associated coefficient-to-function operator products. Total target energy is $V = \operatorname{tr}(\Sigma\mathbb{E}[a(X)a(X)^\top])$. Thus Σ_{learn} describes what the training observations identify, while H describes what the deployed peer responses expose. Their joint effect determines adaptation value. With a fully revealed target, the first restriction disappears but the state constraint does not, as in theorem 1.

A.7 Why stored responses do not determine necessary state

For a fixed positive semidefinite training Gram matrix K and $\tau = n\lambda > 0$, KRR stores

$$\alpha = (K + \tau I)^{-1}Y. \tag{35}$$

The inverse $Y = (K + \tau I)\alpha$ recovers every fitting response in exact arithmetic. By contrast, the deployed predictor uses α only through its inner products with query kernel vectors. These two operations can require different amounts of retained information.

For example, take $k(z, z') = 1$, so $K = \mathbf{1}\mathbf{1}^\top$, and let $Y \in \{0,1\}^n$. The map $Y \mapsto \alpha$ is injective and contains n bits when Y is uniform. Yet for every deployment query,

$$\widehat{y}(z) = \mathbf{1}^\top\alpha = \frac{\sum_i Y_i}{n+\tau}. \tag{36}$$

The entire prediction function is exactly specified by the integer $\sum_i Y_i \in \{0,\dots,n\}$, using $\lceil \log_2(n+1)\rceil$ bits with n, τ public. The reduced state preserves the prediction exactly for every query. Thus an injective fitting representation can contain substantially more target information than an equivalent prediction function requires. With a general kernel, retaining fewer spectral coefficients instead trades prediction error against state size, as examined in section C.6.

B EXPERIMENTAL DETAILS

This section describes the response representation, adapter families, state encodings, and statistical comparisons used in the language-model experiments. The exploratory scalar-response studies and post hoc analyses use the designs specified in sections C.4 and C.6.

B.1 MODELS AND RESPONSE REPRESENTATION

Table 4 lists the model checkpoints and their roles. The alternative peer set also serves as the secondary target set. In the cross-role comparison, the primary targets supply peer responses for the secondary targets. Every comparison excludes the target from its own peer set. The accompanying implementation pins model and tokenizer revisions in `configs/llm_revisions.json`.

Table 4: **Models used as targets and peers.** Secondary targets also form the alternative peer set. Cross-role comparisons use the primary targets as peers for the secondary targets.

Role	Model identifier
Reference peer	`allenai/Olmo-3-7B-Instruct`
Reference peer	`mistralai/Ministral-3-8B-Instruct-2512-BF16`
Reference peer	`Qwen/Qwen3-8B`
Primary target	`Qwen/Qwen3.5-9B`
Primary target	`ibm-granite/granite-3.3-8b-instruct`
Primary target	`google/gemma-4-12B-it`
Secondary target / alternative peer	`mistralai/Mistral-7B-Instruct-v0.3`
Secondary target / alternative peer	`tiiuae/Falcon3-7B-Instruct`
Secondary target / alternative peer	`Qwen/Qwen2.5-7B-Instruct`

Prompts present numbered options and request one answer code without explanation. We use official chat templates, disable optional thinking, and score the first answer-code position. Each selected code occupies one token in every tokenizer. Normalizing logits over these codes defines the response distribution. This measures relative preferences among the options, excluding probability mass assigned to other tokens and subsequent generation.

For m options, let $\ell \in \mathbb{R}^m$ contain the answer-code logits. Let $H_m \in \mathbb{R}^{(m-1)\times m}$ have orthonormal Helmert contrast rows, with $H_m\mathbf{1} = 0$. Adapters predict the coordinates $y = H_m\ell$ and reconstruct probabilities as

$$\widehat{p} = \mathrm{softmax}(H_m^\top \widehat{y}), \qquad p = \mathrm{softmax}(\ell). \tag{37}$$

Softmax is invariant to a common logit shift, so the contrast representation removes one redundant coordinate while preserving the response distribution. Adapters minimize squared fitting error in these coordinates; validation and test errors are TV between the reconstructed distributions. Convex mixing therefore combines contrast coordinates before conversion to probabilities.

Responses were collected with BF16 model weights on NVIDIA GeForce RTX 5090 GPUs, using a 4,096-token context limit and random seed 20260722. Software and model revisions are fixed across response collection. The accompanying implementation specifies the environment and dependencies.

B.2 DATASETS AND PARTITIONS

MMLU-Pro examples come from the benchmark's test split, HellaSwag examples from its validation split, and RewardBench examples from its filtered split. RewardBench is formatted as a choice between two candidate responses. These source collections are partitioned again into fitting, validation, and evaluation sets for the replacement task.

We retain examples compatible with every tokenizer and the context limit, then reserve complete domains for OOD evaluation. Domains are MMLU-Pro categories, HellaSwag activity labels, and RewardBench subsets. Exact normalized-text duplicates and candidate near-duplicates form connected clusters: candidates use 64 MinHash values in eight bands over word 5-grams, with Jaccard

similarity at least 0.85. Clusters are disjoint across fitting, validation, and IID test partitions. This controls textual duplication, not all semantic overlap. Option permutations are consistent across models; table 5 gives sample sizes.

Table 5: **Response samples for each model.** Out-of-domain (OOD) evaluation holds out complete domains; the other three partitions contain disjoint text-duplicate clusters.

Task	Options	Fit	Validation	IID test	OOD test	OOD domains
MMLU-Pro	10	2,048	512	1,024	1,053	1
HellaSwag	4	2,048	512	1,024	1,070	29
RewardBench	2	1,024	256	512	647	3

Target responses provide the supervision. Benchmark labels are used in task construction and stratification, while TV measures reproduction of the target's answer distribution. Agreement with the target can therefore include agreement with its mistakes.

B.3 Adapter families

For linear, RFF, KRR, and MLP adapters, we concatenate the selected peer contrasts and standardize them using fitting-set means and standard deviations; a near-zero standard deviation is replaced by one. Normalizers and feature maps depend only on peer responses. Gaussian-kernel bandwidth uses the median of a deterministic sample of peer-feature distances. Convex mixing instead uses the unstandardized peer contrasts.

Convex mixing. Nonnegative peer weights constrained to sum to one minimize squared fitting error. The resulting quadratic problem is solved with simplex constraints. Quantized variants use a combinatorial encoding of the simplex weights.

Linear and random-feature regression. Linear ridge regression includes an intercept and uses centered normal equations with regularization $n\lambda I$. RFF applies a fixed Gaussian Fourier map to the standardized inputs before ridge regression. Frequencies and phases are generated from the public seed. We consider $16, 32, 64, 128, 256, 512$, and 1024 random features.

Kernel ridge regression. For a peer Gram matrix K and target-coordinate matrix Y, the full-precision fit is $\alpha = (K + n\lambda I)^{-1}Y$. Encoded representations store coefficients in the peer-derived eigenbasis and retain ranks $8, 16, 32, 64, 128, 256, 512, 1024$, or the full admissible rank. Each truncated or quantized candidate is evaluated through its decoded prediction function. These ranks describe the fitted kernel representation; the behavioral covariance in theorem 3 instead describes variation across targets and training observations.

Multilayer perceptron. The MLP has one tanh hidden layer of width 32, 64, or 128 and a linear output layer. Adam uses learning rate 10^{-3}, batch size at most 256, and at most 300 epochs. Ten percent of the fitting partition is used for early stopping, with patience 20 and tolerance 10^{-5}. The separate validation set selects the final candidate. All weights and biases enter the retained state; deterministic rescaling is absorbed into the decoded parameters.

Linear, RFF, and KRR candidates use 17 regularization values logarithmically spaced from 10^{-8} to 10^{2}. The MLP regularization values are $10^{-5.5}$, 10^{-3}, and $10^{-0.5}$.

B.4 Encoding target-specific state

The state contains a 22-byte header identifying the encoding format, adapter family, selected peers, regularization, rank, coordinate dimensions, precision, bias, and payload length. Matrix coefficients use $1, 2, 3, 4, 6, 8, 12, 16$, or 32 bits, together with one FP32 scale per column. Unquantized candidates use FP16, FP32, or FP64. Simplex weights use a composition code. The evaluated predictor decodes exactly this representation.

The bit count includes all these fields and excludes at most seven terminal byte-padding bits. Since the main state caps are byte-aligned, the padded files also fit their caps. The 848-bit example is byte-aligned before padding. Public context comprises the peer-only fitting-response table, normalizers,

eigenvectors, feature-map seeds, and fixed encoder and decoder programs. It is shared across targets and can be large. The measured quantity is the additional state required for a particular target, conditional on that context.

B.5 Validation-based selection

We consider zero state or caps 2^b bits for $b \in \{5, 7, 9, 10, 11, 13, 14, 15, 17, 18, 19, 20, 21\}$ and $q \in \{1, 2, 3\}$ peer calls. For each cap and call count, validation TV selects among feasible candidates, jointly choosing peers, regularization, model size, and encoding. The test set evaluates this choice after selection is complete. A larger cap enlarges the candidate set; the selected state may use substantially fewer bits than allowed.

The three-call comparisons query all three peers. At smaller call counts, the selected subset is included in the encoded state. A predetermined peer or fixed equal-peer mean requires zero target-specific selection bits. Small peer weights alone do not reduce the call count. Infeasible candidates are omitted: for example, no MLP in the candidate grid fits the smallest displayed MMLU-Pro cap. Consequently, there is no three-task MLP mean at that cap.

B.6 Component interventions

Input controls replace peer features before fitting and repeat validation selection under the same state cap. Row shuffling breaks query–response alignment; random addresses and constant features remove informative peer responses. The input-text control uses an L2-normalized TF-IDF representation with at most 256 vocabulary features, fitted on the fitting partition only. These interventions compare the specified feature representations and fitted adapters.

Adapter exchanges instead hold the task and peers fixed and substitute the full-precision state fitted to another target with the same output dimension. Neutral-state interventions have different meanings across families: zero KRR coefficients give a uniform probability vector, whereas equal convex weights average the peer contrasts. We therefore use matched target exchanges to study target specificity. A separate channel-permutation control changes its permutation between data partitions and hence tests an inconsistent interface; it is excluded from the main input comparison. For the one-dimensional RewardBench contrast, that permutation is the identity.

B.7 Uncertainty estimates

Paired 95% intervals use 2,000 percentile-bootstrap draws over text-duplicate item clusters, resampling the same clusters for both predictions. They are pointwise intervals conditional on fitted adapters, without correction for multiple comparisons or repeated-training uncertainty. OOD summaries average within domains before resampling domain blocks. MMLU-Pro has only one held-out domain, so its OOD results are descriptive. Means across target–task combinations summarize the listed models, not a sampled model population.

C Additional Results

C.1 Full-precision and compact replacement

Table 6 compares fitted adapters at their original numerical precision, including the public equal-peer mean. KRR and RFF have the lowest mean errors, but the best family varies across targets and tasks. Table 7 compares each full-precision KRR with the adapter selected under 1,024 bits, including the actual encoded sizes.

The compact adapter is selected jointly with its regularization and retained rank. Its test error can therefore be lower than that of the full-precision fit. The comparison measures the performance attainable by the selected compact representation, rather than an isolated effect of rounding coefficients. The small HellaSwag error changes and larger MMLU-Pro changes show that the amount of useful adaptation is task-dependent.

Table 6: **Replacement error at full precision.** In-distribution test TV with three reference peers. The final column is a paired 95% interval for the KRR-minus-convex difference over held-out query units. Mean denotes equal mixing of peer contrasts.

Target	Task	Mean	Convex	Linear	RFF	KRR	MLP	KRR−Convex CI
Qwen	MMLU-Pro	0.4167	0.3631	0.3538	0.3485	0.3460	0.3448	[-0.0225, -0.0115]
Qwen	HellaSwag	0.2062	0.2003	0.2036	0.1917	0.1902	0.1926	[-0.0132, -0.0072]
Qwen	RewardBench	0.1702	0.1474	0.1507	0.1439	0.1433	0.1459	[-0.0109, 0.0024]
Gemma	MMLU-Pro	0.5128	0.5223	0.4927	0.4708	0.4692	0.4853	[-0.0612, -0.0445]
Gemma	HellaSwag	0.2778	0.2898	0.2051	0.1939	0.1933	0.1918	[-0.1117, -0.0818]
Gemma	RewardBench	0.1577	0.1589	0.1533	0.1494	0.1494	0.1481	[-0.0280, 0.0093]
Granite	MMLU-Pro	0.5189	0.5341	0.5288	0.5218	0.5176	0.5292	[-0.0240, -0.0090]
Granite	HellaSwag	0.2973	0.2881	0.2958	0.2728	0.2748	0.2681	[-0.0190, -0.0075]
Granite	RewardBench	0.2390	0.2211	0.2273	0.2221	0.2217	0.2224	[-0.0044, 0.0054]

Table 7: **Target-specific state can be reduced substantially.** KRR selected under 1,024 bits compared with the full-precision FP64 fit. Both state counts include decoding metadata. ΔTV is compact minus full precision; negative values favor the compact adapter.

Target	Task	Full TV	Compact TV	Compact bits	Full bits	ΔTV
Qwen	MMLU-Pro	0.3460	0.3702	896	1,179,824	+0.0242
Qwen	HellaSwag	0.1902	0.1931	368	393,392	+0.0029
Qwen	RewardBench	0.1433	0.1422	976	65,712	-0.0011
Gemma	MMLU-Pro	0.4692	0.5128	896	1,179,824	+0.0436
Gemma	HellaSwag	0.1933	0.1951	848	393,392	+0.0018
Gemma	RewardBench	0.1494	0.1426	976	65,712	-0.0068
Granite	MMLU-Pro	0.5176	0.5324	896	1,179,824	+0.0148
Granite	HellaSwag	0.2748	0.2758	560	393,392	+0.0009
Granite	RewardBench	0.2217	0.2236	304	65,712	+0.0019

C.2 Budget dependence for individual targets and tasks

Figure 9 separates the task means in the main text into their nine constituent comparisons. All five adapter families use three peers and the same response partitions. The horizontal axis is the permitted target-state budget. Lines connect evaluated caps; repeated points occur when an increased cap leaves the validation-selected adapter unchanged.

C.3 Training data and generalization across domains

Retained state and target training data constrain different parts of replacement. We fit adapters on deterministic nested prefixes of 128, 256, 512, 1,024, or all available examples, selecting candidates on validation data at each size. These comparisons use three peers and FP64 states without a finite state cap. Figure 10 reports IID and held-out-domain TV; RewardBench has 1,024 fitting examples in total.

The held-out-domain results assess transfer beyond the fitting distribution. They do not follow from the fixed-distribution characterization in lemma 2. Increasing training coverage and increasing retained state address different limitations; the finite-state construction removes training uncertainty to isolate the latter.

C.4 Adapter sensitivity in scalar-response tasks

An exploratory study examines scalar responses from visual, textual, and synthetic targets. The visual study uses 250 fitting and 250 evaluation inputs, six intervention strengths, and seven regularization values. The textual study uses 100 fitting and 100 evaluation inputs and seven regularization values. The synthetic study uses eight peers, 240 fitting and 320 evaluation inputs, ten seeds, and eleven regularization values. Table 8 averages the residuals over these settings. The visual average thus covers 42 parameter combinations rather than independent repetitions.

The robust visual target has convex residual 0.268040 and KRR residual 0.016066, moving from the largest to the smallest residual among the six visual targets. Several other targets favor convex mixing. The textual exact-copy condition has zero convex residual but positive regularized KRR residual;

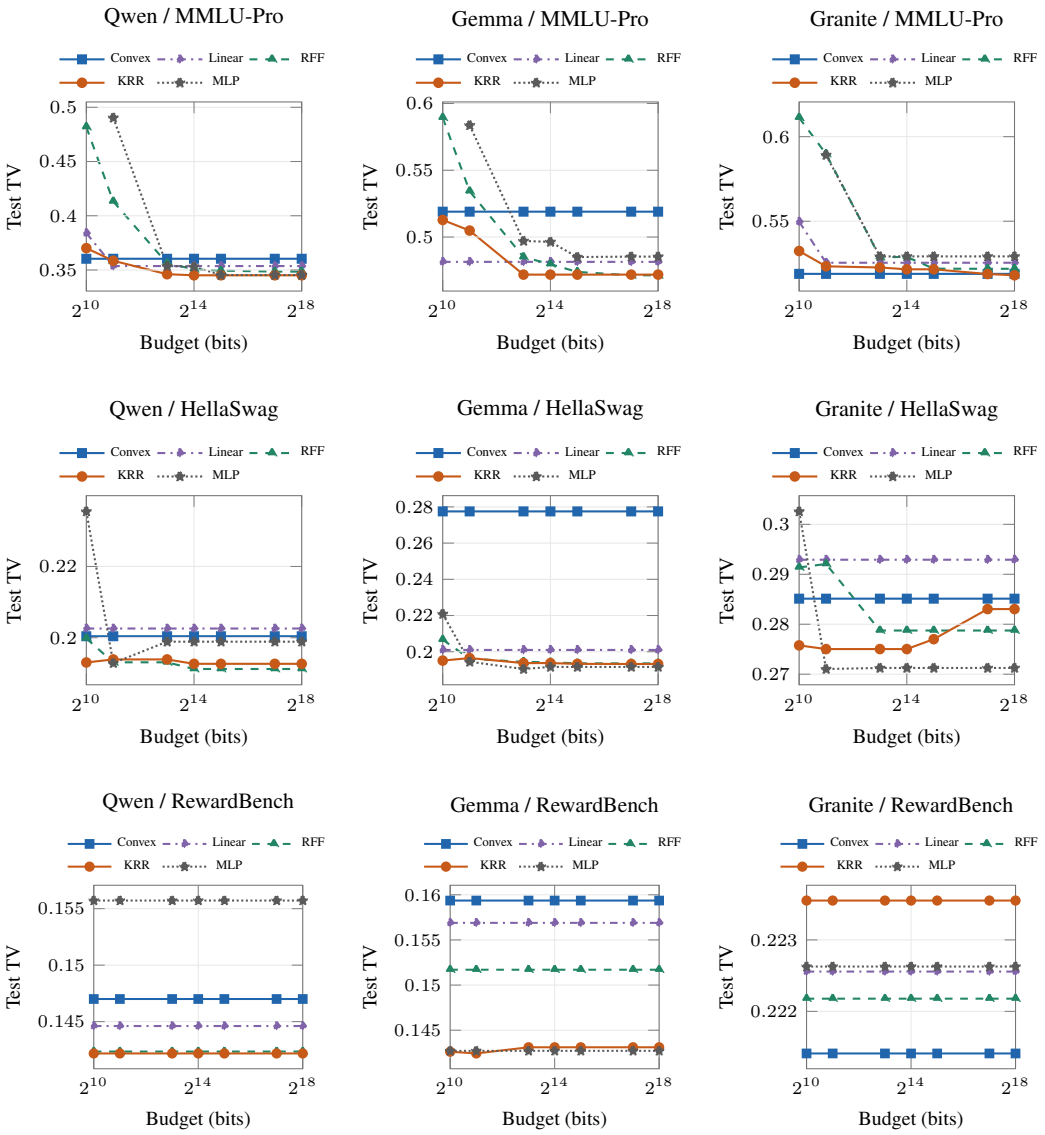

Figure 9: **Adapter rankings vary across targets, tasks, and budgets.** Columns are Qwen, Gemma, and Granite; rows are MMLU-Pro, HellaSwag, and RewardBench. Each point evaluates the validation-selected adapter under its state cap. Missing points indicate that no candidate in that family fits the budget. All comparisons use three peer calls.

the two non-copy textual conditions also favor convex mixing. In synthetic data, convex mixing is better in the convex-hull condition, whereas KRR is better for the nonlinear target. Together, these comparisons show how the fitted function class changes apparent replacement difficulty.

Only aggregate residuals are available for these exploratory studies. They are used to examine adapter sensitivity, separately from the validation-selected language-model comparisons and their paired uncertainty estimates.

C.5 Exact replacement and retained target information

Controlled target families separate exact prediction from adaptation cost. A predetermined peer copy needs one call and no target-specific state. A fixed-length code selecting among 64 behaviors that differ on queries of positive probability needs at least six bits, and their index suffices for exact

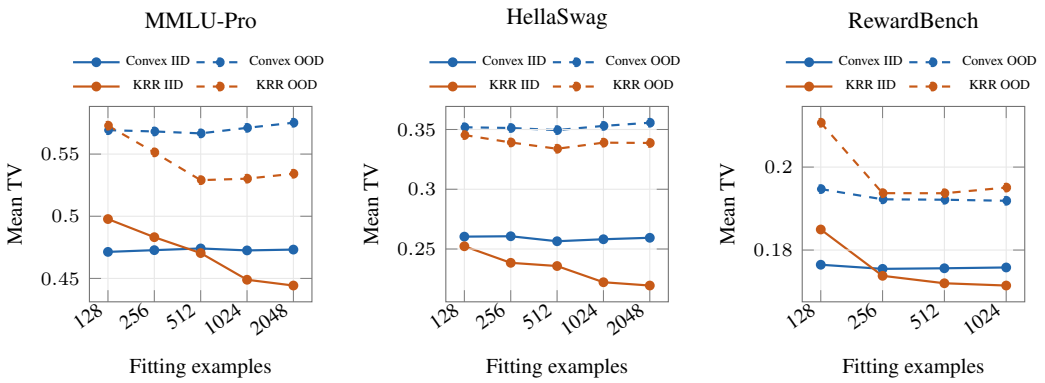

Figure 10: **Replacement depends on training coverage and the query distribution.** Mean TV over the three primary targets as the number of fitting examples increases. Solid lines evaluate in-distribution queries; dashed lines evaluate held-out domains. These domain means are descriptive, including the single held-out MMLU-Pro domain.

Table 8: **The preferred adapter depends on the target response.** Mean scalar-response residuals in the exploratory visual, textual, and synthetic studies. These values average the study-specific parameter settings and are on a different scale from the language-model TV measurements.

Study	Target / condition	Convex residual	KRR residual
Synthetic	Convex hull	0.002442	0.018420
Synthetic	Nonlinear target	0.448066	0.184814
Text	Different architecture	0.159883	0.194138
Text	Different parameters	0.200849	0.244503
Text	Exact peer copy	0.000000	0.003429
Vision	ConvNeXt-Tiny	0.111063	0.120126
Vision	EfficientNet-B0	0.097345	0.121001
Vision	MobileNetV3-Small	0.151754	0.170589
Vision	ResNet-18	0.146455	0.176395
Vision	ResNet-50	0.175598	0.171152
Vision	Robust ResNet-50	0.268040	0.016066

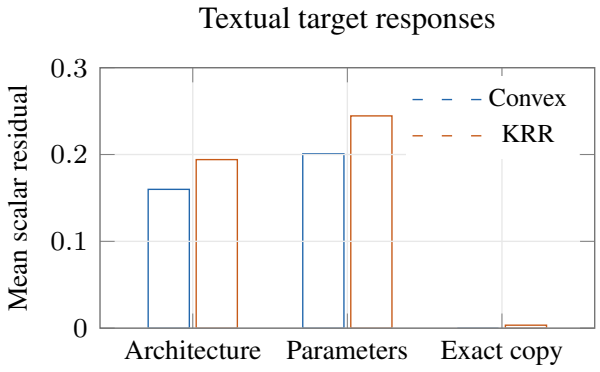

Figure 11: **Direct reuse can outperform nonlinear fitting.** Convex mixing has lower scalar-response residual in each textual condition, including exact reproduction of a peer copy. Values and averaging settings are given in table 8.

replacement. A 512-entry independent binary table can instead be reproduced by retaining all 512 entries. Zero error alone therefore does not specify how much target-dependent state is used.

Fitted KRR also separates information retention from prediction. Its FP64 coefficients recover the fitting-response coordinates to numerical precision through $(K + n\lambda I)\alpha$. The deployment predictor

instead uses the query kernel and α. Consequently, the error of that predictor on fitting examples and the reconstruction error of the algebraic inverse measure different quantities. Section A.7 gives an exact example in which an injective fitted state admits a much smaller prediction-equivalent representation.

There is also a simple parameterization invariance. Scaling both K and λ by $c > 0$ changes α to α/c. Scaling the query kernel by the same factor leaves every prediction unchanged. Thus the numerical value of $1/\lambda$ depends on kernel normalization, whereas the evaluated prediction function is invariant. Explicit state encodings provide a comparable adaptation resource across these parameterizations.

C.6 Predictive state and simple calibration

The following post hoc analyses examine two possible reasons why a fitted adapter may be larger or more flexible than its predictions require: redundant spectral coefficients and simple calibration of peer responses. They use the existing response data and are separate from the budget-based candidate selection in the main study.

Spectral truncation. We refit KRR at its selected regularization and retain the smallest candidate rank whose validation TV is within 0.005 of the full-rank value. Across the nine comparisons in table 9, the coefficient payload decreases by factors of 32–256 and mean test TV increases by 0.00217. At the same time, applying the original algebraic inverse to the truncated coefficients gives substantially poorer recovery of the fitting responses. The loss in this explicit reconstruction is therefore much larger than the change in held-out prediction. Payload ratios concern coefficients only; the complete-state ratios in table 7 also include metadata.

Table 9: **Prediction can survive loss of exact response reconstruction.** Post hoc spectral truncation selected by a validation-TV tolerance of 0.005. Full and truncated TV evaluate test replacement; inverse TV evaluates fitting-response recovery through the original algebraic inverse. The full-precision inverse-TV baseline is below 2×10^{-15} in every row. Payload ratio measures the reduction in coefficient storage.

Target	Task	Rank	Full TV	Truncated TV	Inverse TV	Payload ratio
Qwen	MMLU-Pro	64	0.3460	0.3487	0.3315	32×
Gemma	MMLU-Pro	64	0.4692	0.4716	0.4474	32×
Granite	MMLU-Pro	64	0.5176	0.5227	0.5020	32×
Qwen	HellaSwag	8	0.1902	0.1921	0.2095	256×
Gemma	HellaSwag	8	0.1933	0.1952	0.2189	256×
Granite	HellaSwag	64	0.2748	0.2743	0.2643	32×
Qwen	RewardBench	16	0.1433	0.1457	0.1226	64×
Gemma	RewardBench	8	0.1494	0.1505	0.1455	128×
Granite	RewardBench	8	0.2217	0.2242	0.2302	128×

Nonnegative calibration. Removing the sum-to-one constraint from the three convex weights gives a three-scalar nonnegative least-squares adapter. Its mean test TV is 0.2892, compared with 0.3028 for convex mixing and 0.2784 for full-precision KRR. It accounts for approximately 55.6% of the mean convex-to-KRR improvement. Thus simple scaling explains an appreciable portion of the benefit attributed to a more flexible mapping. These three fitted real coefficients have not been evaluated with a finite-bit encoding.

The two analyses concern predictive performance and an explicit reconstruction method. They support the distinction between a fitted representation and the behavior it implements; they provide neither a minimum-state estimate nor a guarantee against other reconstruction methods.