TRAINING SELECTS AMONG BEHAVIORALLY EQUIVALENT COMPUTATIONS IN IN-CONTEXT DICTIONARY LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Behavioral agreement with a known algorithm does not show which computation a network implements. We make this question exact in in-context complete dictionary learning, where a global inverse, a support-masked inverse, and a support-gated local decoder are provably identical on every training query and differ only beyond the sparse set. A sparse-trained Transformer does not learn the global inverse: fed queries denser than its training set it returns sparse-shaped outputs, and its derivative on held-out directions lies nearest the gated-local decoder, the minimum-norm endpoint of the equivalent class. Training design, not the task, sets where in that class the network lands. At matched sparse-task accuracy, removing the auxiliary support losses of the standard objective moves the learned map toward the support-masked inverse and widening the training query support moves it toward the global one. One support step or a 0.5% dense-query admixture removes most of the sparse-to-dense performance gap; the admixture acts within hundreds of steps on a trained network and reverts once withdrawn, while the derivative on sparse queries keeps its position in the class. As dictionaries grow, errors concentrate in selecting the support rather than in decoding on it. When supervised behavior leaves the computation unidentified, training design selects the extension.

1 INTRODUCTION

When a neural network's behavior matches a familiar algorithm, what evidence shows that it actually implements that algorithm? In-context learning makes this question especially sharp. Transformers trained on regression and related statistical tasks often reproduce the input-output behavior of least squares, gradient descent, and other recognizable estimators (Garg et al., 2022; von Oswald et al., 2023; Akyürek et al., 2023; Bai et al., 2023). These results reveal what Transformers can learn in context, not how the learned network computes its answer.

Behavior is an incomplete certificate because different computations can agree everywhere the training distribution reaches and differ elsewhere. Interpretability therefore faces a prior question: *what evidence is sufficient to identify the computation a network actually implements?* The in-context learning literature records this question as unfinished: its founding work frames the goal as understanding the learning algorithm encoded by the model (Garg et al., 2022), follow-ups describe first steps toward a complete characterization (Akyürek et al., 2023), and controlled-organism studies still list the implemented algorithm as future work (Gopalani et al., 2024).

We make the question answerable in complete dictionary learning (DL). Sparse autoencoders already treat a linear dictionary with a sparse encoder as a working model of learned representations (Bricken et al., 2023; Templeton et al., 2024; Huben et al., 2024; Lieberum et al., 2024). Our controlled setting asks which sparse encoder a Transformer learns in context. Each episode hides a fresh square dictionary $\boldsymbol{A}^\natural$ and sparse context codes. The Transformer sees only their mixtures and must decode a new query, with supervision on the query code but never on the dictionary or context codes. The generator and canonicalization define the reference inverse $\boldsymbol{P}^\natural = (\boldsymbol{A}^\natural)^{-1}$, which the context identifies when the codes are sufficiently scattered (App. A). Identifying the dictionary does not identify the computation: with $\boldsymbol{P}^\natural$ pinned, infinitely many maps still agree with it on every sparse

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.

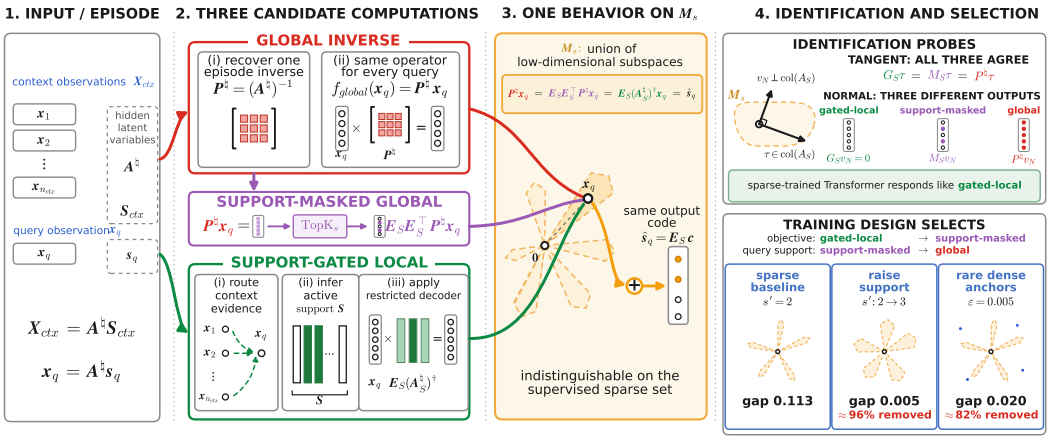

Figure 1: **One behavior, three function extensions.** Global, support-masked, and gated-local decoders agree on every valid sparse query yet differ beyond the supervised subspaces. Held-out probes place the standard sparse-trained Transformer nearest gated-local, the reference with no response off the sparse set (Sec. 3). Removing support-side losses selects support-masked; broader support under pure MSE moves the nearest reference to global (k=5, matched fresh cosine; Sec. 4). App. N gives a panel-by-panel guide.

query and differ elsewhere, so the ambiguity that structure removes from the latent factors reappears one level up, in the map the network implements. Unlike ordinary out-of-distribution questions, this ambiguity comes with a candidate set: three maps that are each the canonical member of a computation class and are exactly equal on the entire training distribution, so asking which one training selects is well posed.

Three computations solve this task identically on every sparse query: solve only the selected subproblem, apply one global inverse and mask its output, or apply the global inverse unmasked. They differ only in how they extend beyond the supervised subspaces, so that is where we look. Probed there, the sparse-trained network sits nearest the gated-local decoder and far from the global inverse, and changing the training design moves it toward the other two at matched task accuracy.

Contributions.

- **Behavioral success does not identify the computation, even when the dictionary is identified.** Global, support-masked, and gated-local decoders agree on every sparse query, and each is the canonical member of a computation class: the unique query-linear extension, the Jacobian of a TopK sparse-autoencoder encoder, and the minimum-Frobenius-norm exact-interpolant Jacobian (Sec. 2).
- **Sparse training lands nearest the gated-local decoder, far from the global inverse.** On held-out directions the standard model's derivative is nearest gated-local at every $k \geq 5$: it carries the right action on the active directions and little response on the normal ones, partway toward support-masked at small k. Fed denser queries than it was trained on, the network returns sparse-shaped outputs while a dense-trained control tracks the target (Sec. 3).
- **Training design switches the computation beneath matched task behavior.** At matched sparse accuracy, removing the support-side losses moves the nearest reference from gated-local to support-masked and widening the query support moves it toward global. The extension has a near part and a far part: a small dense-query admixture removes most of the sparse-to-dense gap within hundreds of steps on a trained network and reverts once withdrawn, while the derivative on sparse queries keeps its position (Sec. 4).
- **Failure at larger k is a selection failure rather than an information failure.** As dictionaries grow, support selection collapses while decoding on the correct support stays high, and a classical algorithm recovers the dictionary from the same contexts, as a support-gated computation predicts (Sec. 5).

2 SPARSE BEHAVIOR LEAVES THREE COMPUTATIONS EQUIVALENT

2.1 IN-CONTEXT COMPLETE DICTIONARY LEARNING

Classical dictionary learning models each observation as a sparse combination of the columns, or atoms, of an unknown dictionary: the data satisfy $\boldsymbol{X} = \boldsymbol{A}^\natural \boldsymbol{S}^\natural$, where neither the dictionary $\boldsymbol{A}^\natural$ nor the sparse codes $\boldsymbol{S}^\natural$ are known. We use the complete case, with square invertible $\boldsymbol{A}^\natural \in \mathbb{R}^{k\times k}$, because it supplies the canonical linear query map $\boldsymbol{P}^\natural = (\boldsymbol{A}^\natural)^{-1}$. Overcomplete decoding is itself nonlinear and offers no such linear reference (App. A). The volume formulation of Hu & Huang (2023b) gives a global recovery guarantee under the sufficiently-scattered condition, so when that condition holds the observed context determines the latent factors up to permutation and scaling; canonicalization fixes the remaining atom order and signs. Identifiability conditions (including the instance-level check at k=3), the fixed-cardinality caveat, the estimator, and canonicalization are in Apps. A, H, and B.

Each episode draws a fresh dictionary, exact-s context codes (exactly s nonzero coordinates each), and query codes. The model sees only $\boldsymbol{X}_{\mathrm{ctx}} = \boldsymbol{A}^\natural \boldsymbol{S}_{\mathrm{ctx}}$ and query observations $\boldsymbol{x}_q = \boldsymbol{A}^\natural \boldsymbol{s}_q$, and predicts $\boldsymbol{s}_q$. Unless noted, $k \in \{3, 5, 8, 12\}$, s=2, and n_{ctx}=128. The main model is a six-layer, eight-head bidirectional Transformer (Vaswani et al., 2017) with d_{model}=256. Two linear heads read each final query state, a coefficient head (the primary target) and an auxiliary support head. The coefficient head is not gated by the support head, so the gated-local extension found in Sec. 3 is a property of the learned function, not of the architecture (details in Apps. D, E, and G).

2.2 BEHAVIORAL EQUIVALENCE

For support S, let $\boldsymbol{A}^\natural_S$ contain the active dictionary columns and let $\boldsymbol{E}_S$ embed an s-vector into $\mathbb{R}^k$.

Definition 1 (Sparse query set and candidate computations)**.** For $1 \le s \le k$, define

$$M_s = \bigcup_{|S|=s} \mathrm{col}(\boldsymbol{A}^\natural_S). \tag{1}$$

For any $\boldsymbol{x} \in \mathbb{R}^k$, let $S_{\mathrm{mask}}(\boldsymbol{x})$ contain the s largest-magnitude coordinates of $\boldsymbol{P}^\natural \boldsymbol{x}$, and let $S_{\mathrm{local}}(\boldsymbol{x})$ minimize $\mathrm{dist}(\boldsymbol{x}, \mathrm{col}(\boldsymbol{A}^\natural_S))$ over $|S| = s$; both use fixed deterministic tie rules. The three completed query maps are

$$f_{\mathrm{global}}(\boldsymbol{x}) = \boldsymbol{P}^\natural \boldsymbol{x}, \tag{2}$$

$$f_{\mathrm{mask}}(\boldsymbol{x}) = \boldsymbol{E}_{S_{\mathrm{mask}}} \boldsymbol{E}_{S_{\mathrm{mask}}}^\top \boldsymbol{P}^\natural \boldsymbol{x}, \tag{3}$$

$$f_{\mathrm{local}}(\boldsymbol{x}) = \boldsymbol{E}_{S_{\mathrm{local}}} (\boldsymbol{A}^\natural_{S_{\mathrm{local}}})^\dagger \boldsymbol{x}. \tag{4}$$

Proposition 1 (Behavioral equivalence on sparse queries)**.** *For every query $\boldsymbol{x} = \boldsymbol{A}^\natural_S \boldsymbol{c} \in M_s$, the completed maps obey*

$$f_{\mathrm{global}}(\boldsymbol{x}) = f_{\mathrm{mask}}(\boldsymbol{x}) = f_{\mathrm{local}}(\boldsymbol{x}) = \boldsymbol{P}^\natural \boldsymbol{x} = \boldsymbol{E}_S \boldsymbol{c}. \tag{5}$$

At every exact-s query, one with all s coefficients nonzero, both selectors recover its unique support S.

The identity follows from $\boldsymbol{A}^\natural_S = \boldsymbol{A}^\natural \boldsymbol{E}_S$, $\boldsymbol{P}^\natural \boldsymbol{A}^\natural_S = \boldsymbol{E}_S$, and $(\boldsymbol{A}^\natural_S)^\dagger \boldsymbol{A}^\natural_S = \boldsymbol{I}_s$ (App. B). For $s < k$ the three maps agree on every supervised query yet differ beyond M_s; at $s = k$ they coincide everywhere (the degenerate geometries where two branches coincide are in Apps. B, F.2). The selectors make the masked and local references full query maps, whose local branches Sec. 3 evaluates at the true support. Throughout, a computation is the query map itself, including its extension beyond M_s; identifying one means pinning down this map to the resolution the data allow, not reconstructing the circuitry that implements it. Two probes measure the extension, the derivative at sparse queries (its near part) and the response beyond M_s (its far part), which Sec. 4.3 shows training selects separately.

2.3 WHAT SPARSE BEHAVIOR DETERMINES

Lemma 1 (The global inverse is the unique query-linear solution)**.** *Let $\boldsymbol{A}^\natural$ be invertible and $1 \le s \le k$. Any single linear map $\boldsymbol{L} : \mathbb{R}^k \to \mathbb{R}^k$ that agrees with the sparse-code target on M_s satisfies $\boldsymbol{L} = \boldsymbol{P}^\natural$.*

The proof is a spanning argument (App. C). For $s < k$, nonlinear alternatives remain: adding a monomial that couples $s+1$ code coordinates leaves every s-sparse output unchanged, and its square also leaves every s-sparse Jacobian unchanged (App. C). Identification is therefore relative to a candidate class and a probe resolution: within the query-linear class sparse behavior determines the computation uniquely, in the unrestricted nonlinear class it does not. When the context codes are sufficiently scattered, the context identifies the latent operator, but sparse supervision does not identify its computational extension (Corollary 1); zero population risk constrains the predictor only where the training-query distribution has support (Proposition 2, App. C).

Each reference is canonical for a computation class. Lemma 1 makes $\boldsymbol{P}^\natural$ the unique query-linear extension. A shared linear encoder $\boldsymbol{W}$ followed by TopK/thresholding has local Jacobian $\boldsymbol{E}_S\boldsymbol{E}_S^\top\boldsymbol{W}$ at fixed selected support. Exact sparse agreement pins $\boldsymbol{W}$ to $\boldsymbol{P}^\natural$ for $s \geq 2$, making the support-masked field the class-canonical Jacobian (App. B). Proposition 3 makes the gated decoder's Jacobian the unique minimum-Frobenius-norm Jacobian, at almost every sparse query, among exact interpolants. For every exact interpolant differentiable at such a query,

$$\boldsymbol{J}\boldsymbol{\Pi}_S = \boldsymbol{E}_S(\boldsymbol{A}_S^\natural)^\dagger, \qquad \|\boldsymbol{J}\|_F^2 \geq \|(\boldsymbol{A}_S^\natural)^\dagger\|_F^2, \tag{6}$$

so sparse data pin the tangent block, the gated decoder attains equality by zeroing the free normal block, and the global inverse pays a strict normal-direction excess whenever $s < k$ (proof in App. I).

The results sit in a layered picture: without structure, latent factors of a nonlinear mixture are not identifiable; the sparse-code structure fixes $\boldsymbol{P}^\natural$; and the ambiguity reappears in the computation, where it is pointwise rather than distributional and therefore decidable by measurement (App. C). Which member a finite-budget learner selects is an optimization question that identifiability theory does not answer, and it is the question of Secs. 3 and 4.

3 A HELD-OUT DIAGNOSTIC IDENTIFIES THE EXTENSION

Ordinary sparse outputs cannot distinguish the references, so we probe their responses where they disagree: the derivative at an isolated query, the finite response along a held-out direction, and the output on queries denser than the training set.

3.1 THE ISOLATED-QUERY DERIVATIVE IS NEAREST GATED-LOCAL AND FAR FROM GLOBAL

For a fixed context and one independently forwarded query, define

$$\boldsymbol{J}(\boldsymbol{x}_q) = \frac{\partial \widehat{\boldsymbol{s}}_q}{\partial \boldsymbol{x}_q}. \tag{7}$$

Across episodes (one is drawn in Fig. 2) the standard model's Jacobian is far from $\boldsymbol{P}^\natural$ in ambient space yet matches the restricted inverse $(\boldsymbol{A}_S^\natural)^\dagger$ along the active directions, the ones the sparse task probes, and its effective rank stays near s rather than k (Table 3): it applies the right inverse on the active subproblem without carrying the normal-direction action of $\boldsymbol{P}^\natural$. Among the three references that makes it nearest gated-local from $k{=}5$ on (Table 1), and its squared Jacobian norm stays within a few times the variational floor of Sec. 2.3 while the global inverse rises to 157 times that floor at $k{=}12$ (Fig. 9, Table 4). The ranking predicts fresh behavior: on a held-out normal direction where the references disagree, gated-local has the lowest finite response error at $k = 5, 8, 12$ (Table 5).

Projecting the derivative onto the two-parameter family through the three references places the dense-trained control on the global vertex and the standard model between gated-local and support-masked at $k{=}5$ ($\alpha{=}0.37$), closer to the vertex at $k{=}8$ and on it at $k{=}12$, with no component toward global at any k and a residual outside the family that grows with k (Table 8). The references are landmarks for this position: a normal block of the network's size with random orientation is nearest gated-local on almost every query, and the network's block is in addition oriented toward support-masked far beyond that size-matched random block at $k{=}5$ and 8 (Table 9, App. F.2). Sec. 4 asks what moves this position.

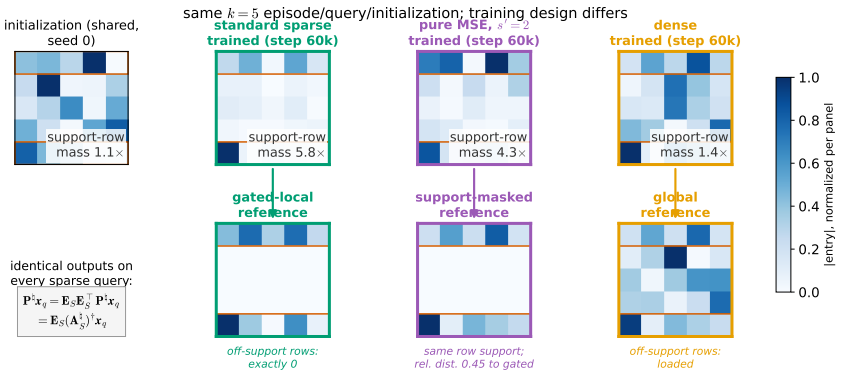

Figure 2: **The selection map in real matrices.** On one shared k=5 episode, query, and initialization, trained Jacobians (top) align by shape with gated-local, support-masked, and global references (bottom) under standard sparse, pure-MSE s'=2, and dense training. Orange lines mark true-support rows; panels are normalized per shape. The support-row mass ratio separates the support-dependent pair from global, while normal directions separate masked from local (Table 1). Cross-episode results are in Table 3.

Table 1: **Three-reference Jacobian distance** on the primary evaluation bank, in units of the gated-local field norm (medians over correct-support queries, so the distances characterize the decoder given selection, Apps. F.1, F.2; lower is closer). The support-dependent references nearly tie at $k = 3$; gated-local separates from $k = 5$ onward. A second bank gives the same ordering at different levels (Table 4).

k	global	support-masked	gated-local
3	1.01	**0.49**	0.51
5	2.20	1.04	**0.79**
8	4.22	2.16	**1.09**
12	12.63	4.74	**1.60**

3.2 Dense queries expose the extension

A global inverse maps a dense observation to its dense code, while the two support-dependent decoders are capped near the trained sparsity. Fed queries denser than anything it was trained on, the sparse-trained network answers with sparse-shaped outputs: its effective output dimension saturates far below the target at every k, while the dense-trained control tracks the target on the same probe (Fig. 7; no single linear map explains the dense behavior, App. F).

The standard model's separation from the global regime replicates at a second width, under three non-Gaussian coefficient laws, under corrected momentum SGD, and at sparsity s=3 (Apps. G, K).

4 Training design selects among the extensions

The diagnostic turns extension choice into an experimental outcome. We vary two training axes: support-side objective terms and the query-support distribution, with everything else held fixed (controls and matched-source verification in App. F.2).

4.1 The training distribution sets the extension

The three primary training settings differ only in the distribution of the query support size s': sparse training fixes s'=2, mixed training draws each query's s' uniformly from $\{2, \ldots, k\}$, and dense training fixes s'=k. All models are scored on the same fixed sparse and dense evaluation sets with every query forwarded independently (App. G); the primary metric is the direct-query cosine drop from sparse to dense evaluation, the sparse-to-dense gap: a large drop indicates a local extension, a near-zero drop that dense queries are decoded as well as sparse ones.

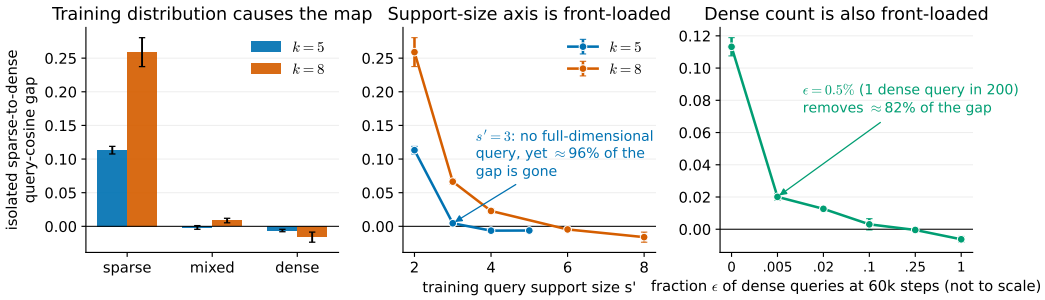

Figure 3: **A strongly front-loaded causal response.** Isolated sparse-to-dense gap under three training-query interventions. Left: sparse training (all queries s'=2) versus mixed (s' uniform on $\{2,\ldots,k\}$) and dense (s'=k). Middle: the fixed training support size raised stepwise from s'=2. One step (s'=3) removes about 96% of the gap at k=5. Right: a 0.5% dense admixture at k=5 removes about 82%; the fraction is a mixture weight (about 6×10^5 dense queries over training, App. G). Error bars are SD over five seeds (endpoints) or three matched seeds (intermediate settings).

Under the same architecture, sparse training yields a nonlinear extension while mixed and dense training yield maps close to one linear operator (Table 13, App. G), so the sparse setting's local map is a selection rather than a capacity limit. All three settings retain high sparse-query accuracy, and mixed training still contains sparse queries, so sparse inputs alone do not force the local regime: the computation changes beneath behavior the task barely distinguishes.

Both response curves in Fig. 3 are front-loaded. Raising fixed support from s'=2 to s'=3 removes about 96% of the k=5 sparse-to-dense gap, although that distribution contains no full-dimensional query; at k=8 the same step is intermediate, so the transition is graded rather than binary. A 0.5% dense-query admixture, making only that fraction of the k=5 training queries dense, removes about 82% of the gap; the dose response and the count-versus-fraction accounting are in App. G, and Sec. 4.3 shows the same admixture acting on an already trained network.

The same response appears with query interaction masked out, without the support-head BCE, and under a pure coefficient-MSE objective with matched expected training energy (App. G), so the selection tracks the support structure of the supervised coefficient targets themselves.

4.2 Objective and query support move the extension along two axes

The reference diagnostic resolves what the gap alone cannot. We vary two parts of the training design one at a time at k=5, the smallest dictionary where the diagnostic separates all three references, with architecture, data, optimizer, budget, and seeds matched and fresh sparse-query cosine held level (Table 6).

The objective. Removing the support-side losses moves the nearest reference from gated-local to support-masked, the encoder form a TopK sparse autoencoder computes, and removing the support BCE alone does the same (App. F.2), at the same sparse-task accuracy. The gap experiments above use the standard objective; the support step below holds pure MSE fixed so that the two axes are separated.

The query support. Holding pure MSE fixed, one energy-matched support step moves the nearest reference from support-masked to global. Projecting a trained map's held-out response onto the two-parameter family through the three references gives two coordinates, α from gated-local toward support-masked and β from support-masked toward global (Fig. 4). The support step lands at an interior point, $(\alpha,\beta)=(0.88,0.57)$: the nearest reference changes without reaching the global vertex, which dense-query training does reach. Both moves change the direction of the off-set response and not only its size: a size-matched control built from each arm's own Jacobians stays nearest gated-local and shifts its reference distances by a fraction of the network's (Table 10). At k=8 the same two changes move the same two coordinates in the same directions by smaller amounts, under the paper's dictionary sampler and under a strictly conditioned one, and again beyond the size-matched control, with no arm moving past support-masked (Table 7, App. F.2).

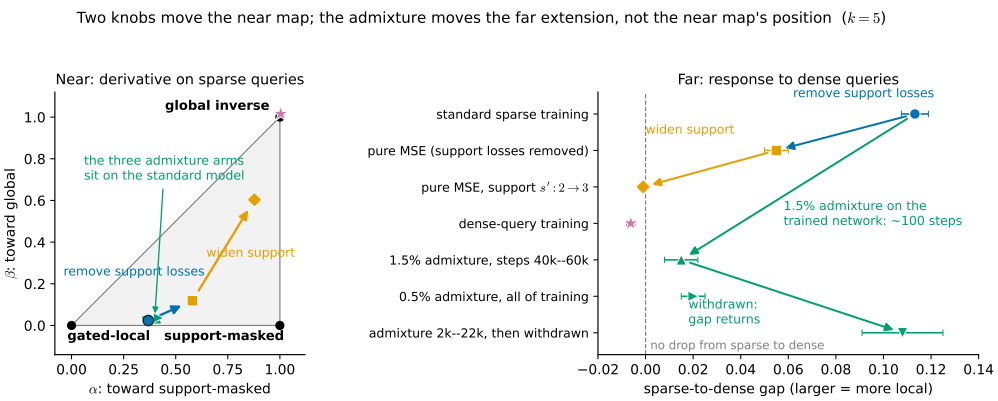

Figure 4: **Two knobs move the near map; the admixture moves the far extension and leaves the near map's position in place** (k=5; the same seven training arms appear in both panels). Left: each arm's sparse-query derivative projected onto the family through the three references, α toward support-masked and β toward global (all-query medians, App. F.2). Removing the support-side losses (blue arrow) and widening the training support (orange arrow) move the point; dense-query training (star) lands on the global vertex, which calibrates the coordinates; the three admixture arms (green) sit on the standard model. Right: the same arms' sparse-to-dense gap (seed mean $\pm$ SD), zero meaning no drop from sparse to dense queries. The two design steps also lower the gap, while the admixture lowers only the gap: applied to a trained network at a 1.5% in-window fraction (the same dense-query count as the 0.5% arm) it converts the far extension within about a hundred steps, and once withdrawn the gap returns (per-arm values in App. F.2).

4.3 The choice is fast and reversible

Along training checkpoints, the sparse baseline has formed its gap by the earliest checkpoint we measure, the density-three setting never does, and the dense-admixture setting is already intermediate; no setting migrates late (App. J). The choice is not fixed by that early formation (Fig. 5). Giving a fully trained k=5 sparse network the admixture's number of dense queries only during steps 40k–60k (a 1.5% in-window fraction) reduces its gap from 0.117 to 0.015, the level the uniform admixture reaches, and half of that reduction is in place 100 steps after the admixture starts. Giving them only during steps 2k–22k reaches that level inside the window, after which sparse-only training returns the gap to its baseline over several thousand steps. Through onset and return the Jacobian on sparse queries keeps its projected family coordinates while the dense-query gap moves severalfold. The Jacobian itself drifts under training, in the normal block that sparse data do not pin: converted and unconverted networks end no farther apart than one undosed run drifts, on three seeds, and the dose changes the size of that drift by at most 0.06 against a drift of 0.6 (in units of the gated-local field norm; App. J). The extension therefore has two parts that two probes measure (Fig. 4): the derivative at sparse queries, which the objective and support axes move, and the response to dense queries, which the admixture moves; the two respond to different knobs. The training distribution converts a formed extension as readily as it shapes a forming one, and the converted extension is held by the admixture rather than retained.

5 Errors concentrate in support selection as dictionaries grow

A support-gated computation carries a selection bottleneck, so as k grows its errors should increasingly concentrate there, and they do: support selection falls from 0.98 at k=3 to 0.58 at k=12 while decoding conditional on the correct support stays at 0.83 or above, and oracle masking recovers much of the loss on wrong-selection queries (Table 14, Fig. 8). The information itself remains available: the volume-minimization algorithm of Hu & Huang (2023b) recovers every episode of the matched evaluation from exactly the contexts on which the Transformer degrades (Fig. 6; it is not compute-matched, and it rules out missing context information). The decomposition locates the failure; a matched query-linear control shows a similar pattern (App. H), so the identification rests on Secs. 3 and 4.

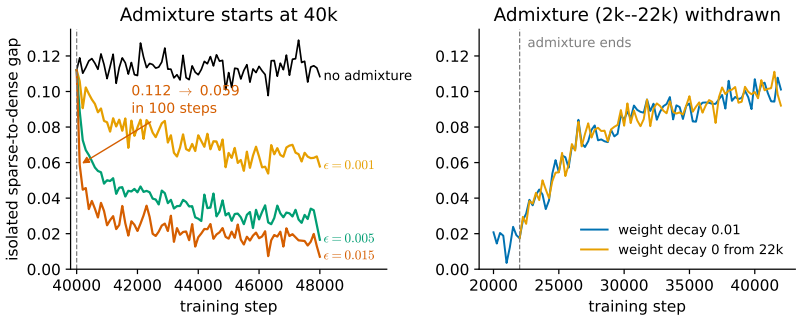

Figure 5: **The far extension converts within hundreds of steps and reverts once the admixture is withdrawn** (k=5, seed 0, 100-step checkpoints). Left: from a 40k sparse-trained checkpoint, dense queries are mixed into training at fraction ϵ; the sparse-to-dense gap falls within the first 100 steps and settles by dose. Right: an admixture applied during steps 2k–22k and then withdrawn; the gap returns to the sparse baseline over several thousand steps, with or without weight decay. The sparse-query Jacobian's family coordinates stay put along both trajectories (Fig. 10); three-seed endpoints are in App. J.

6 Related work

In-context algorithms, data distributions, and controlled organisms. Transformers learn least squares, sparse regression, gradient descent, and algorithm selection in context (Garg et al., 2022; von Oswald et al., 2023; Akyürek et al., 2023; Bai et al., 2023); we ask what is selected when canonical and alternative computations are behaviorally identical. Different algorithms appear across hyperparameters (Zhong et al., 2023), and data distributions, curricula, and competitive dynamics change in-context strategies (Chan et al., 2022; Lee et al., 2025; Li et al., 2023b; Park et al., 2025). Closest to our message, Raventós et al. (2023) find a pretraining task-diversity threshold below which a Transformer behaves as a Bayesian estimator under the narrow pretraining prior and above which it matches ridge regression, and Wu et al. (2024) prove that few tasks suffice for single-layer linear attention to approach ridge; in both, the candidate algorithms differ on the supervised distribution itself. Our candidates are exactly equal on the full supervised output over a continuous query set, so only their extension distinguishes them. Kwon et al. (2026) show, in single-layer linear attention, that union-of-subspaces task vectors let the learned map generalize to subspaces of zero training mass; our sparse queries also form a union of subspaces, but the dictionary is hidden and changes every episode, and the question is which equivalent map a nonlinear model selects. Modular arithmetic, matrix completion, and Othello supply ground truth for grokking and circuit studies (Power et al., 2022; Nanda et al., 2023; Gopalani et al., 2024; Gopalani & Hu, 2025; Li et al., 2023a; Olsson et al., 2022; Wang et al., 2023), and control tasks and amnesic interventions separate encoded structure from causal use (Hewitt & Liang, 2019; Elazar et al., 2021). Méloux et al. (2025) find systematic non-uniqueness of mechanistic explanations by enumeration over small networks. Here the candidates are proved equivalent against a canonical reference, and held-out probes rank them.

Dictionary learning, identifiability, and implicit bias. Dictionary learning has a long lineage (Olshausen & Field, 1997; Aharon et al., 2006a); App. A reviews its identifiability results, and we use the global result and estimator of Hu & Huang (2023b) for the canonical episode reference. In-context sparse linear regression keeps prompt covariates observed (Chen et al., 2024), whereas our dictionary is hidden; sparse coding has also served architecturally in compositional in-context learning (Chen et al., 2025). A linear encoder followed by TopK, the standard TopK-SAE form (Gao et al., 2025) taken here in the complete case, has a support-masked-form Jacobian at fixed selected support (the selected rows of its own encoder, pinned to the inverse by exact sparse agreement, App. B), motivating the middle reference; here sparse coding is the task itself. Nonlinear ICA and disentanglement leave latent representations ambiguous without extra structure (Hyvärinen & Pajunen, 1999; Locatello et al., 2019); the sparse-code structure supplies exactly that, identifies the latent factors when codes are sufficiently scattered, and the ambiguity reappears one level up, in the computation (App. C). Underspecification studies find many predictors with equally strong held-out

performance, the choice sensitive to the seed (D'Amour et al., 2022). In our setting the candidates are provably characterized and the selection is seed-stable and set by the training design.

7 Conclusion and discussion

Complete dictionary learning supplies a canonical inverse, yet sparse supervision leaves three computations equivalent, and a sparse-trained Transformer does not learn the inverse: held-out probes place it nearest the gated-local vertex of the equivalent class, and dense queries expose the local extension. The objective and the query support move the extension along two named directions at matched task accuracy, and a dense admixture converts its far part within hundreds of steps and, once withdrawn, reverts over thousands, while the derivative on sparse queries keeps its position. As dictionaries grow, errors concentrate in selecting the support. When behavior leaves the computation unidentified, training design selects the extension.

A claim that a network implements a known algorithm is a claim about behavior unless it comes with candidates that agree on the training distribution and a probe where they disagree. Here the candidates are theorems and the probe ranks them, so the question has a measured answer. The resulting non-identifiability is structured rather than accidental. The selected extension is stable across seeds and moves with the objective and the query distribution, so underspecification here is a design variable rather than merely seed noise. And the far extension is held by the current training distribution, converting within hundreds of steps and returning once the dense queries stop, so a mechanistic description of off-support behavior taken at one checkpoint describes the network under its current data, while the near map's position in the class is what persists. In our complete-DL organism the TopK encoder form used by sparse autoencoders is one canonical member of the class, and the support terms of the objective decide whether the in-context learner sits near it. The recipe transfers: construct candidates that agree on the training distribution, probe where they disagree, and intervene on training.

Outlook. These results are for complete dictionaries, where a global inverse exists to serve as the canonical candidate; overcomplete dictionaries and trained sparse autoencoders, where it does not, are where the recipe goes next. What remains open here is a question of optimization rather than of interpretation: Proposition 3 says why the gated-local decoder is canonical, not why training selects it (App. I), and an account of that selection would predict which design change moves the extension.

AI USE STATEMENT

This work was directed by the authors: the research question, the choice of setting, every experimental and control design, every claim made or withheld, and the final interpretation were decided by the authors after discussion with generative AI tools. Within that process we used generative AI tools as an assistant for implementing methods, including the data generator and evaluation code; for helping refine the conceptual framework of candidate computations and their references; for formulating mathematical claims and assisting in the writing and checking of proofs; for proposing candidate hypotheses and preregistered predictions, which the authors selected and fixed before each run; for giving feedback on experiments and controls; and for suggesting interpretations of results, which were treated as hypotheses until checked against the artifacts. We have not used generative AI tools to generate data (all data are sampled from the stochastic model of App. D by our own code), to clean or reformat datasets, or for translation; qualitative or thematic data analysis is not applicable to this work. Additionally, we used generative AI tools to draft and edit parts of the paper, create and modify figures, search and summarize related literature, format references, and suggest the paper's structure. We have reviewed all AI-assisted work: code was checked by independent recomputation and planted-defect tests; every proposition was checked by exact-arithmetic falsifiers and audited manually; cited passages were verified against the sources; and every reported number traces to a versioned artifact and producer script. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The appendix specifies data generation, architecture, optimization, metrics, interventions, budgets, seeds, controls, and checkpoint policy (App. M), and provides complete proofs of the mathematical claims (Apps. B, C, and I). All data are synthetic, and these specifications support independent reconstruction of the study design and reported estimands. The result manifest maps every reported quantity to model tag, checkpoint/seed list, evaluation context, aggregation rule, source JSON, and producer script. No checkpoint was selected separately for a reported metric.

REFERENCES

Michal Aharon, Michael Elad, and Alfred Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11): 4311–4322, 2006a. URL https://doi.org/10.1109/tsp.2006.881199.

Michal Aharon, Michael Elad, and Alfred M. Bruckstein. On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them. *Linear Algebra and its Applications*, 416(1):48–67, 2006b. URL https://doi.org/10.1016/j.laa.2005.06.035.

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? Investigations with linear models. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2211.15661.

Sanjeev Arora, Rong Ge, Tengyu Ma, and Ankur Moitra. Simple, efficient, and neural algorithms for sparse coding. In *Proceedings of the 28th Conference on Learning Theory (COLT)*, volume 40 of *Proceedings of Machine Learning Research*, pp. 113–149. PMLR, 2015. URL https://proceedings.mlr.press/v40/Arora15.html.

Yu Bai, Fan Chen, Huan Wang, Caiming Xiong, and Song Mei. Transformers as statisticians: Provable in-context learning with in-context algorithm selection. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://doi.org/10.52202/075280-2498.

David G. T. Barrett and Benoit Dherin. Implicit gradient regularization. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://arxiv.org/abs/2009.11162.

José M. Bioucas-Dias. A variable splitting augmented Lagrangian approach to linear spectral unmixing. In *2009 First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote*

Sensing (WHISPERS), pp. 1–4, 2009. URL https://doi.org/10.1109/whispers.2009.5289072.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nicholas L Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Stephanie C. Y. Chan, Adam Santoro, Andrew K. Lampinen, Jane X. Wang, Aaditya K. Singh, Pierre H. Richemond, James L. McClelland, and Felix Hill. Data distributional properties drive emergent in-context learning in transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL https://doi.org/10.52202/068431-1371.

Tsung-Han Chan, Chong-Yung Chi, Yu-Min Huang, and Wing-Kin Ma. A convex analysis-based minimum-volume enclosing simplex algorithm for hyperspectral unmixing. *IEEE Transactions on Signal Processing*, 57(11):4418–4432, 2009. URL https://doi.org/10.1109/tsp.2009.2025802.

Wei Chen, Jingxi Yu, Zichen Miao, and Qiang Qiu. In-context compositional learning via sparse coding transformer. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://doi.org/10.52202/085713-1484.

Xingwu Chen, Lei Zhao, and Difan Zou. How transformers utilize multi-head attention in in-context learning? A case study on sparse linear regression. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL https://doi.org/10.52202/079017-3799.

Jeremy E. Cohen and Nicolas Gillis. Identifiability of complete dictionary learning. *SIAM Journal on Mathematics of Data Science*, 1(3):518–536, 2019. URL https://doi.org/10.1137/18m1233339.

Maurice D. Craig. Minimum-volume transforms for remotely sensed data. *IEEE Transactions on Geoscience and Remote Sensing*, 32(3):542–552, 1994. URL https://doi.org/10.1109/36.297973.

Alexander D'Amour, Katherine Heller, Dan Moldovan, et al. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 23(226):1–61, 2022. URL https://jmlr.org/papers/v23/20-1335.html.

Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175, 2021. URL https://doi.org/10.1162/tacl_a_00359.

Xiao Fu, Kejun Huang, Nicholas D. Sidiropoulos, and Wing-Kin Ma. Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications. *IEEE Signal Processing Magazine*, 36(2):59–80, 2019. URL https://doi.org/10.1109/msp.2018.2877582.

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

Shivam Garg, Dimitris Tsipras, Percy Liang, and Gregory Valiant. What can transformers learn in-context? A case study of simple function classes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL https://doi.org/10.52202/068431-2217.

Quan Geng and John Wright. On the local correctness of ℓ_1-minimization for dictionary learning. In *2014 IEEE International Symposium on Information Theory (ISIT)*, pp. 3180–3184, 2014. URL https://doi.org/10.1109/isit.2014.6875421.

Pulkit Gopalani and Wei Hu. What happens during the loss plateau? Understanding abrupt learning in transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/4f06c73c45d2625f0e687f7e6a206332-Abstract-Conference.html.

Pulkit Gopalani, Ekdeep Singh Lubana, and Wei Hu. Abrupt learning in transformers: A case study on matrix completion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL https://doi.org/10.52202/079017-1747.

Rémi Gribonval and Karin Schnass. Dictionary identification: Sparse matrix-factorization via ℓ_1-minimization. *IEEE Transactions on Information Theory*, 56(7):3523–3539, 2010. URL https://doi.org/10.1109/tit.2010.2048466.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016. URL https://arxiv.org/abs/1606.08415.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2733–2743, 2019. URL https://doi.org/10.18653/v1/d19-1275.

Jingzhou Hu and Kejun Huang. Identifiable bounded component analysis via minimum volume enclosing parallelotope. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2023a. URL https://doi.org/10.1109/icassp49357.2023.10095905.

Jingzhou Hu and Kejun Huang. Global identifiability of ℓ_1-based dictionary learning via matrix volume optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023b. URL https://doi.org/10.52202/075280-1568.

Jingzhou Hu and Kejun Huang. Complex bounded component analysis: Identifiability and algorithm. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6680–6684, 2024. URL https://doi.org/10.1109/icassp48485.2024.10446804.

Jingzhou Hu and Kejun Huang. Linearized ADMM for simplicial and nonnegative component analyses. In *2025 59th Asilomar Conference on Signals, Systems, and Computers*, pp. 436–439, 2025. URL https://doi.org/10.1109/ieeeconf67917.2025.11443634.

Kejun Huang, Xiao Fu, and Nicholas D. Sidiropoulos. Anchor-free correlated topic modeling: Identifiability and algorithm. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016a. URL https://proceedings.neurips.cc/paper/2016/hash/d707329bece455a462b58ce00d1194c9-Abstract.html.

Kejun Huang, Nicholas D. Sidiropoulos, and Athanasios P. Liavas. A flexible and efficient algorithmic framework for constrained matrix and tensor factorization. *IEEE Transactions on Signal Processing*, 64(19):5052–5065, 2016b. URL https://doi.org/10.1109/tsp.2016.2576427.

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

Aapo Hyvärinen and Petteri Pajunen. Nonlinear independent component analysis: Existence and uniqueness results. *Neural Networks*, 12(3):429–439, 1999. URL https://doi.org/10.1016/S0893-6080(98)00140-3.

Soo Min Kwon, Alec S. Xu, Can Yaras, Laura Balzano, and Qing Qu. Out-of-distribution generalization of in-context learning: A low-dimensional subspace perspective. In *Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 300 of *Proceedings of Machine Learning Research*, pp. 4285–4293, 2026. URL https://proceedings.mlr.press/v300/kwon26a.html.

Jin Hwa Lee, Andrew Kyle Lampinen, Aaditya K. Singh, and Andrew M. Saxe. Distinct computations emerge from compositional curricula in in-context learning. In *Workshop on Spurious Correlation and Shortcut Learning: Foundations and Solutions (SCSL), ICLR*, 2025. URL https://openreview.net/forum?id=oo5TNikeJl.

Kenneth Li, Aspen K. Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Emergent world representations: Exploring a sequence model trained on a synthetic task. In *International Conference on Learning Representations (ICLR)*, 2023a. URL https://arxiv.org/abs/2210.13382.

Yingcong Li, Kartik Sreenivasan, Angeliki Giannou, Dimitris Papailiopoulos, and Samet Oymak. Dissecting chain-of-thought: Compositionality through in-context filtering and learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023b. URL https://doi.org/10.52202/075280-0966.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on Gemma 2. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 278–300. Association for Computational Linguistics, 2024. URL https://doi.org/10.18653/v1/2024.blackboxnlp-1.19.

Chia-Hsiang Lin, Wing-Kin Ma, Wei-Chiang Li, Chong-Yung Chi, and ArulMurugan Ambikapathi. Identifiability of the simplex volume minimization criterion for blind hyperspectral unmixing: The no-pure-pixel case. *IEEE Transactions on Geoscience and Remote Sensing*, 53(10):5530–5546, 2015. URL https://doi.org/10.1109/tgrs.2015.2424719.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pp. 4114–4124, 2019. URL https://proceedings.mlr.press/v97/locatello19a.html.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019. URL https://arxiv.org/abs/1711.05101.

Chao Ma and Lexing Ying. On linear stability of SGD and input-smoothness of neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/8c26d2fad09dc76f3ff36b6ea752b0e1-Abstract.html.

Maxime Méloux, Silviu Maniu, François Portet, and Maxime Peyrard. Everything, everywhere, all at once: Is mechanistic interpretability identifiable? In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/51157713f1f1f572a05647e1aef1b1ed-Abstract-Conference.html.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2301.05217.

José M. P. Nascimento and José M. Bioucas-Dias. Vertex component analysis: A fast algorithm to unmix hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing*, 43(4):898–910, 2005. URL https://doi.org/10.1109/tgrs.2005.844293.

Bruno A. Olshausen and David J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37(23):3311–3325, 1997. URL https://doi.org/10.1016/s0042-6989(97)00169-7.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli,

Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread; arXiv preprint arXiv:2209.11895*, 2022. URL https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

Greg Ongie, Rebecca Willett, Daniel Soudry, and Nathan Srebro. A function space view of bounded norm infinite width ReLU nets: The multivariate case. In *International Conference on Learning Representations (ICLR)*, 2020. URL https://arxiv.org/abs/1910.01635.

Core Francisco Park, Ekdeep Singh Lubana, and Hidenori Tanaka. Competition dynamics shape algorithmic phases of in-context learning. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/a635abfdd007c9a698410c8bbe5ebc33-Abstract-Conference.html.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022. URL https://arxiv.org/abs/2201.02177.

Allan Raventós, Mansheej Paul, Feng Chen, and Surya Ganguli. Pretraining task diversity and the emergence of non-Bayesian in-context learning for regression. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://doi.org/10.52202/075280-0626.

Daniel A. Spielman, Huan Wang, and John Wright. Exact recovery of sparsely-used dictionaries. In *Proceedings of the 25th Annual Conference on Learning Theory (COLT)*, volume 23 of *Proceedings of Machine Learning Research*, pp. 37.1–37.18. PMLR, 2012. URL https://proceedings.mlr.press/v23/spielman12.html.

Ju Sun, Qing Qu, and John Wright. Complete dictionary recovery over the sphere I: Overview and the geometric picture. *IEEE Transactions on Information Theory*, 63(2):853–884, 2017a. URL https://doi.org/10.1109/tit.2016.2632162.

Ju Sun, Qing Qu, and John Wright. Complete dictionary recovery over the sphere II: Recovery by Riemannian trust-region method. *IEEE Transactions on Information Theory*, 63(2):885–914, 2017b. URL https://doi.org/10.1109/tit.2016.2632149.

Yuchen Sun and Kejun Huang. Improved identifiability and sample complexity analysis of complete dictionary learning. In *2024 IEEE 13th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pp. 1–5. IEEE, 2024. URL https://doi.org/10.1109/sam60225.2024.10636510.

Yuchen Sun and Kejun Huang. Global identifiability of overcomplete dictionary learning via L1 and volume minimization. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/84409c45a0defe347c895b004b1c675b-Abstract-Conference.html.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. *Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 35151–35174. PMLR, 2023. URL https://proceedings.mlr.press/v202/von-oswald23a.html.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2211.00593.

Yu Wang, Siqi Wu, and Bin Yu. Unique sharp local minimum in ℓ_1-minimization complete dictionary learning. *Journal of Machine Learning Research*, 21(63):1–52, 2020. URL https://jmlr.org/papers/v21/19-169.html.

Jingfeng Wu, Difan Zou, Zixiang Chen, Vladimir Braverman, Quanquan Gu, and Peter L. Bartlett. How many pretraining tasks are needed for in-context learning of linear regression? In *International Conference on Learning Representations (ICLR)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/fa3c139cf8084de7bfd944f1c90c8695-Abstract-Conference.html.

Siqi Wu and Bin Yu. Local identifiability of ℓ_1-minimization dictionary learning: a sufficient and almost necessary condition. *Journal of Machine Learning Research*, 18(168):1–56, 2018. URL https://jmlr.org/papers/v18/16-119.html.

Yuexiang Zhai, Zitong Yang, Zhenyu Liao, John Wright, and Yi Ma. Complete dictionary learning via ℓ_4-norm maximization over the orthogonal group. *Journal of Machine Learning Research*, 21 (165):1–68, 2020. URL https://jmlr.org/papers/v21/19-755.html.

Ziqian Zhong, Ziming Liu, Max Tegmark, and Jacob Andreas. The clock and the pizza: Two stories in mechanistic explanation of neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://doi.org/10.52202/075280-1186.

A Background: identifiability and the reference estimator

Our setting has a genuine ground truth: each episode's data $\boldsymbol{X}_{\mathrm{ctx}} = \boldsymbol{A}^{\natural} S_{\mathrm{ctx}}$ is generated from an actual dictionary $\boldsymbol{A}^{\natural}$ and codes, so $\boldsymbol{A}^{\natural}$ (after the canonicalization of App. B) and its inverse $\boldsymbol{P}^{\natural}$ are well defined objects, not modeling assumptions.

Definition 2 (Essential identifiability, adapted from Hu & Huang (2023b))**.** The model $\boldsymbol{X} = \boldsymbol{A}^{\natural}\boldsymbol{S}$ is *essentially identifiable* (up to signed permutation) if, under a condition on the codes $\boldsymbol{S}$, every optimal solution of the volume-minimization criterion $\min_{\boldsymbol{A},\boldsymbol{S}} |\det \boldsymbol{A}|$ subject to $\boldsymbol{X}{=}\boldsymbol{A}\boldsymbol{S}$ and $\|\boldsymbol{S}_{j,:}\|_1 \le 1$ equals $(\boldsymbol{A}^{\natural}, \boldsymbol{S})$ up to a column permutation and a per-column $\pm$ sign, after the rows of the ground-truth $\boldsymbol{S}$ are rescaled to unit ℓ_1 norm with the scales absorbed into the columns of $\boldsymbol{A}^{\natural}$ (the general statement allows an arbitrary diagonal rescaling). Under identifiability the canonical inverse $\boldsymbol{P}^{\natural} = (\boldsymbol{A}^{\natural})^{-1}$ is the unique recovery target against which the network can be compared.

Remark 1 (Sample-complexity intuition). Volume-minimization identifiability holds when the codes are *sufficiently scattered*. Precisely, rescale each row of the code matrix to unit ℓ_1 norm and form all 2^n signed sums of the codes (one for each sign pattern in $\{\pm 1\}^n$). The condition requires the convex hull of these signed sums to contain the ℓ_2 unit ball, to lie inside the hypercube $[-1, 1]^k$, and to touch the ball's boundary only at $\pm\boldsymbol{e}_1, \ldots, \pm\boldsymbol{e}_k$ (Hu & Huang, 2023b). Under the Bernoulli–Gaussian model with density p, this occurs with high probability once $n_{\mathrm{ctx}} \gtrsim c\, k \log k/(p(1-p))$ (Hu & Huang, 2023b; Sun & Huang, 2024). Our fixed-cardinality context distribution is this Bernoulli–Gaussian model conditioned on having exactly s nonzeros per code, with an additional minimum-magnitude floor on active values. The conditioning makes the support indicators within a code dependent, so the i.i.d. rate does not literally cover it. We use the deterministic sufficiently-scattered theorem as the reference and verify instance-level recoverability directly, by running the volume-minimization estimator on the identical contexts (App. H). At $k{=}3$ the scattering condition itself is verified instance-level: the signed-sum hull equals the zonotope $\sum_i [-\tilde{\boldsymbol{s}}_i, \tilde{\boldsymbol{s}}_i]$, so the condition reduces to $\min_{\|\boldsymbol{u}\|_2=1} \sum_i |\langle \tilde{\boldsymbol{s}}_i, \boldsymbol{u}\rangle| \ge 1$ with equality only at $\pm\boldsymbol{e}_j$, whose minimum is attained at arrangement vertices, giving a finite per-episode check. On 1000 fresh episodes the condition holds on 830 at the training context length $n_{\mathrm{ctx}}{=}128$, rising to $935/986/999$ per thousand at $n_{\mathrm{ctx}}{=}192/256/512$. The estimator recovers the dictionary on every one of these episodes; on the 830 the sufficient condition for uniqueness holds with a clear numerical margin, and the remaining episodes recover without it (the condition is sufficient rather than necessary). We do not

transfer this Bernoulli–Gaussian rate to our fixed-cardinality setting. Instead, we separate two empirical questions in App. H: whether severe target-atom under-coverage hurts individual predictions, and whether the full context is sufficient for the classical estimator. The first effect saturates at the primary context length, and the second is verified directly on identical contexts. A formal fixed-cardinality in-context sample-complexity statement is left to future work.

The same-context oracle we compare against (App. H) solves this volume-minimization criterion by the linearized ADMM (L-ADMM) of Hu & Huang (2023b), which we implement following the published algorithm. ADMM-based splitting is the standard algorithmic framework for constrained factorization criteria of this type (Huang et al., 2016b; Hu & Huang, 2025).

Broader identifiability context. Identifiability results for dictionary learning include ℓ_1 criteria, exact recovery of sparsely-used dictionaries, uniqueness of overcomplete dictionaries, local conditions and local correctness of ℓ_1 minimization, complete-dictionary guarantees, and nonconvex recovery (Gribonval & Schnass, 2010; Spielman et al., 2012; Aharon et al., 2006b; Wu & Yu, 2018; Geng & Wright, 2014; Wang et al., 2020; Cohen & Gillis, 2019; Sun et al., 2017a;b; Zhai et al., 2020; Arora et al., 2015). Modern identifiability theory also extends beyond the complete setting (Sun & Huang, 2025). We use the complete case because it gives the linear reference $\boldsymbol{P}^\natural$ needed for the computation-level test. For observations in $\mathbb{R}^d$ and k atoms, the model is complete when $d = k$ and overcomplete when $k > d$. A tall full-column-rank dictionary ($d > k$) reduces to the square recovery problem after restricting queries to its k-dimensional column space and choosing coordinates there. The reduction does not preserve unrestricted ambient extensions: every $L = A^\dagger + B(I - AA^\dagger)$ agrees on dictionary-generated observations while differing on the orthogonal complement. Thus $d = k$ is a normalization for recovery and for extensions restricted to the data subspace. The genuinely excluded regime is overcomplete $k > d$, where no linear left inverse exists. Several of the local results guarantee only that the ground truth is a sharp local optimum, whereas the volume criterion used here is global. Minimum-volume criteria originate in hyperspectral unmixing, from minimum-volume transforms and simplex algorithms to identifiability without pure pixels (Craig, 1994; Nascimento & Bioucas-Dias, 2005; Chan et al., 2009; Bioucas-Dias, 2009; Lin et al., 2015), and sufficiently-scattered-type conditions extend this identifiability to anchor-free topic models and a broader family of constrained factorizations (Huang et al., 2016a; Fu et al., 2019; Hu & Huang, 2023a; 2024).

B Formal behavioral equivalence

Proposition. Let $\boldsymbol{A}^\natural \in \mathbb{R}^{k\times k}$ be invertible and $S \subseteq [k]$ with $|S| = s$. Write $\boldsymbol{A}^\natural_S \in \mathbb{R}^{k\times s}$ for the active columns and $\boldsymbol{E}_S \in \mathbb{R}^{k\times s}$ for the embedding that places s active coefficients into the k-dimensional code. For any query $\boldsymbol{x} = \boldsymbol{A}^\natural_S \boldsymbol{c}$ with $\boldsymbol{c} \in \mathbb{R}^s$,

$$(\boldsymbol{A}^\natural)^{-1}\boldsymbol{x} = \boldsymbol{E}_S \boldsymbol{E}_S^\top (\boldsymbol{A}^\natural)^{-1}\boldsymbol{x} = \boldsymbol{E}_S (\boldsymbol{A}^\natural_S)^\dagger \boldsymbol{x} = \boldsymbol{E}_S \boldsymbol{c}.$$

Proof. $(\boldsymbol{A}^\natural)^{-1}\boldsymbol{A}^\natural_S = \boldsymbol{E}_S$, so the global output already has support S and masking leaves it unchanged. Since $\boldsymbol{A}^\natural_S$ has full column rank, $(\boldsymbol{A}^\natural_S)^\dagger \boldsymbol{A}^\natural_S = I_s$, proving the local equality. □

Thus the global inverse, support-masked global field $\boldsymbol{E}_S \boldsymbol{E}_S^\top \boldsymbol{P}^\natural$, and support-gated local field $\boldsymbol{E}_S (\boldsymbol{A}^\natural_S)^\dagger$ agree exactly on $\mathrm{col}(\boldsymbol{A}^\natural_S)$. With the TopK and nearest-subspace selectors of Definition 1, these fields define full query maps. At exact-s sparse queries (all s coefficients nonzero) the support is unique and both selectors return it; at lower-sparsity intersections, every zero-distance support yields the same canonical code $\boldsymbol{P}^\natural \boldsymbol{x}$. Off M_s, global retains all inverse rows, support-masked retains only selected rows, and gated-local annihilates the normal complement. At $s = k$ all three coincide everywhere. Output accuracy on valid sparse queries cannot distinguish them.

Shared-encoder TopK class. Let $W \in \mathbb{R}^{k\times k}$ be one linear encoder followed by TopK/thresholding, and suppose its selected support is locally fixed to S. Its Jacobian is $E_S E_S^\top W$. If this decoder agrees exactly with the sparse-code target for every size-s support, with $s \geq 2$, then $W = \boldsymbol{P}^\natural$. Indeed, for any i, j choose a support S containing both (the case $i = j$ only requires $i \in S$). Exactness of selected row i on $x = \boldsymbol{A}^\natural_S c$ gives $w_i^\top a_j = \delta_{ij}$. Hence $W\boldsymbol{A}^\natural = I$ and $W = \boldsymbol{P}^\natural$.

The class-canonical local field is therefore $E_S E_S^\top \boldsymbol{P}^\natural$. The restriction $s \geq 2$ is necessary: at $s = 1$, sparse agreement fixes $w_i^\top a_i = 1$ but leaves the off-diagonal products undetermined. Requiring correct TopK selection over the stated coefficient support can bound their magnitudes, but still does not identify them.

Canonicalization. The factorization $\boldsymbol{X} = \boldsymbol{AS}$ is defined only up to permutation and scaling. Fixing unit-norm columns pins the per-column magnitude and leaves a per-column sign, so the dictionary is identifiable up to a *signed* permutation. We remove this residual ambiguity by sorting the columns of $\boldsymbol{A}^\natural$ by a lexicographic key after making each column's largest-magnitude entry positive, giving a unique $\boldsymbol{A}^\natural$ and hence a uniquely specified $\boldsymbol{P}^\natural$ (ties occur with probability zero under our continuous dictionary distribution). All ground-truth codes and metrics are computed in this canonical frame.

C THE UNIQUE-LINEAR-OPERATOR LEMMA AND THE TRAINING-FREE CONFIRMATION

Population version.

Proposition 2 (What the training-query distribution constrains)**.** *Let μ be the training-query distribution, let $Q = \mathrm{supp}(\mu)$, and let the population target be $T(\boldsymbol{x}) = \boldsymbol{P}^\natural \boldsymbol{x}$. For any measurable predictor $f : \mathbb{R}^k \to \mathbb{R}^k$, zero population squared-error risk, $\mathbb{E}_{\boldsymbol{x}\sim\mu}\|f(\boldsymbol{x}) - T(\boldsymbol{x})\|_2^2 = 0$, implies that $f = T$ μ-almost surely. If $Q \subseteq M_s$ with $s < k$, as in sparse-query training, the predictor remains unconstrained outside M_s, and this is true even in the idealized case $Q = M_s$. If μ instead has a density that is positive almost everywhere on a full-dimensional set U, zero risk constrains the predictor Lebesgue-almost everywhere on U. Neither statement specifies what a finite-budget optimizer selects among the functions that fit the target where μ has support.*

In words: training can only pin down the function where training queries actually go, and a sparse query distribution never visits most of the space.

Explicit nonlinear construction (from Sec. 2.3). When $s < k$, write $\boldsymbol{z} = \boldsymbol{P}^\natural \boldsymbol{x}$ for the code of an observation, choose any index set J with $|J| = s + 1$ and any vector $\boldsymbol{v}$. Then

$$\boldsymbol{h}(\boldsymbol{z}) = \boldsymbol{v} \prod_{j \in J} z_j \tag{8}$$

vanishes on every s-sparse code, since its $|J| = s + 1$ coordinates cannot all be nonzero in an s-sparse vector. Thus $f(\boldsymbol{x}) = \boldsymbol{P}^\natural \boldsymbol{x} + \boldsymbol{h}(\boldsymbol{P}^\natural \boldsymbol{x})$ agrees with the target on all of M_s but differs elsewhere. Scaling $\boldsymbol{v}$ and varying J give infinitely many alternatives. Squaring the monomial removes even the first-order signature: with $p(\boldsymbol{z}) = \prod_{j\in J} z_j$, the map $\tilde{f}(\boldsymbol{x}) = \boldsymbol{P}^\natural \boldsymbol{x} + \boldsymbol{v}\, p(\boldsymbol{P}^\natural \boldsymbol{x})^2$ satisfies $\nabla[p^2] = 2p\,\nabla p = 0$ wherever $p = 0$, hence agrees with the global inverse in both value and Jacobian at every s-sparse code while differing away from M_s. Finite off-manifold probes therefore constrain the extension without certifying uniqueness in the unrestricted class; the identification claims of Secs. 3–4 are relative to the three-reference candidate class and the probe resolution stated there.

Lemma (restated). With $M_s = \bigcup_{|S|=s} \mathrm{col}(\boldsymbol{A}^\natural_S)$ and $\boldsymbol{A}^\natural$ invertible, M_s spans $\mathbb{R}^k$ (for $1 \leq s \leq k$ the union of the column spans contains each atom $\boldsymbol{a}_i$, e.g. via any support $S \ni i$, and the atoms span $\mathbb{R}^k$). Any single linear $\boldsymbol{L}$ with $\boldsymbol{Lx} = \boldsymbol{P}^\natural \boldsymbol{x}$ for all $\boldsymbol{x} \in M_s$ therefore agrees with $\boldsymbol{P}^\natural$ on a spanning set, so $\boldsymbol{L} = \boldsymbol{P}^\natural$. Consequently a query-linear map cannot realize the support-gated local computation. Locality requires nonlinearity.

Corollary 1 (Latent identifiability does not identify a nonlinear computation)**.** *Suppose the context identifies the dictionary up to permutation and scaling and canonicalization fixes $\boldsymbol{A}^\natural$, hence $\boldsymbol{P}^\natural$. Then the latent operator is identified, and the query-linear computational class has the unique solution $\boldsymbol{P}^\natural$ on M_s. For $s < k$, the unrestricted nonlinear class does not: infinitely many functions agree with $\boldsymbol{P}^\natural$ on M_s and differ off it.*

Table 2: **The network deviates from the global inverse implied by its own sparse behavior.** $\widehat{\boldsymbol{P}}$ is a single linear map fit by least squares to the network's own isolated sparse-query input–output pairs, with no dense data: if the network implemented one linear operator, $\widehat{\boldsymbol{P}}$ would reproduce it everywhere. Indeed $\widehat{\boldsymbol{P}}$ approximates $\boldsymbol{P}^{\natural}$ and largely inverts dense inputs, yet the network itself falls below this self-implied linear extension on dense queries, moderately at k=5 and sharply at k=8: its sparse behavior implies an approximate global inverse that its dense behavior does not follow. The score is mean absolute cosine on the active coordinates, isolated protocol.

k	$\|\widehat{\boldsymbol{P}}\boldsymbol{A}^{\natural} - \boldsymbol{I}\|_F/\sqrt{k}$	$\widehat{\boldsymbol{P}}$ on dense	network on dense
5	0.18	0.944	0.830
8	0.27	0.946	0.662

Layers of identifiability. The results of this section sit in a layered picture. (i) *Latent identifiability without structure fails*: for general nonlinear mixtures, observationally equivalent models with entangled latents always exist, so no amount of unlabeled data identifies the factors (Hyvärinen & Pajunen, 1999; Locatello et al., 2019). The equivalence there is distributional. (ii) *Structure restores latent identifiability*: the sufficiently scattered sparse-code condition is exactly such structure, identifying the dictionary and codes up to signed permutation and, after canonicalization, fixing $\boldsymbol{P}^{\natural}$. (iii) *The ambiguity reappears at the computation*: with $\boldsymbol{P}^{\natural}$ identified, the map implementing it is pinned only on M_s, and the equivalence is now pointwise rather than distributional. Infinitely many functions agree with $\boldsymbol{P}^{\natural}$ on every sparse query (the construction above). For analytic deviations, the freedom first appears at interaction order s+1 (Lemma 2) and is eliminated at s'=k, where the pieces become full-dimensional (App. I), and within the query-linear class the solution is unique (Lemma 1). Because the level-(iii) equivalence is pointwise and lives off the data support, it is decidable by measurement, unlike the distributional equivalences of level (i): derivatives, off-manifold probes, and interventions settle it (Secs. 3–4). Which member a finite-budget learner selects is a further, optimization-level question that identifiability theory does not answer. Sec. 4 shows that objective terms and query support select different references, and the 0.5% dense admixture plays an analogous role at this level: a small amount of additional supervision selects among the equivalents.

Training-free extrapolation protocol. For a sparse-trained model and a fixed context we collect the network's outputs on 96 isolated sparse (s=2) queries, fit $\widehat{\boldsymbol{P}}$ by least squares ($\hat{\boldsymbol{s}} = \widehat{\boldsymbol{P}}\boldsymbol{x}$), and report $\|\widehat{\boldsymbol{P}}\boldsymbol{A}^{\natural} - \boldsymbol{I}\|_F/\sqrt{k}$. We then apply $\widehat{\boldsymbol{P}}$ to 96 isolated dense (s'=k) queries and score mean absolute cosine on the active coordinates against the true dense code (all k coordinates are active for dense targets), and we score the network's own dense outputs the same way. Averaged over 24 episodes: $\|\widehat{\boldsymbol{P}}\boldsymbol{A}^{\natural} - \boldsymbol{I}\|/\sqrt{k} = 0.178/0.268$, $\widehat{\boldsymbol{P}}$ dense alignment $0.944/0.946$, network dense alignment $0.830/0.662$ for k=5/8 (isolated single-query forwards, superseding an earlier batched-forward version that overstated the k=5 deviation). The operator implied by the network's sparse behavior approximates the global inverse and extrapolates to dense inputs, while the network falls below that extrapolation outside M_s, which is the measured difference. Table 2 summarizes the result.

D DATA AND TRAINING

Episodes. Each episode samples a fresh $\boldsymbol{A}^{\natural} \in \mathbb{R}^{k\times k}$ with i.i.d. Gaussian entries and ℓ_2-normalized columns. Observation dimension is d=k (complete). The historical implementation uses a nominal condition-number threshold of 5 but stops rejection sampling after 50 replacement rounds, so it does not guarantee $\kappa(\boldsymbol{A}^{\natural}) \le 5$. A 10,000-draw audit of the exact training sampler gives median realized condition numbers $2.94/4.01/11.37/34.42$ at $k = 3/5/8/12$, with threshold pass rates $100\%/99.98\%/35.87\%/0.12\%$. We preserve this realized distribution for checkpoint-compatible evaluation rather than silently changing the generator. Codes have fixed cardinality: exactly s=2 active coordinates per token, active values Gaussian with minimum magnitude $|c_i| \ge 0.3$, noiseless (σ=0). Context length n_{ctx}=128, training queries n_{query}=8. Context codes are never revealed, and the query code is the target. For every evaluation set we regenerate episodes from the distribution used to train the corresponding model, including the capped conditioning sampler, so evaluation matches training. The primary $k = 8$ isolated sparse-to-dense gap remains $0.243/0.257/0.273$ after

stratifying the canonical evaluation set into $\kappa \leq 5$, $5 < \kappa \leq 16$, and $\kappa > 16$. The mixed and dense settings remain near zero in every stratum. Thus ill-conditioned episodes amplify neither the existence nor the direction of the causal contrast. Large-k performance, however, includes a real conditioning burden, and we do not describe $\kappa > 5$ as out-of-distribution.

Model. Bidirectional pre-norm Transformer encoder, d_{model}=256, d_{ff}=1024, 6 layers, 8 heads, GELU activations, dropout 0, and no positional encoding. A two-layer GELU MLP maps each observation to d_{model}; learned type embeddings distinguish CLS, context, query-separator, and query tokens, with learned CLS and query-separator embeddings. In the standard encoder every token can attend every other token: query tokens can read one another, and context/special-token states can read queries. Two linear heads read each final query state, one for support logits and one for coefficients. The coefficient head is *not* gated by the support head.

Objective and optimization. $\mathcal{L} = \text{MSE}(\hat{\boldsymbol{c}}, \boldsymbol{c}) + \lambda_{\text{sup}} \text{BCE}(\text{support logits}, \mathbf{1}_S) + \lambda_0 \,\text{mean}_{b,q,i \notin S_{bq}} |\hat{c}_{bqi}|$, where the last term averages predicted magnitudes over ground-truth off-support coordinates and is zero when none exist. We use λ_{sup}=1 and λ_0=0.01. Unless an experiment states otherwise, we use AdamW (Loshchilov & Hutter, 2019), learning rate 3×10^{-4}, weight decay 0.01, gradient-norm clipping at 1, a 2000-step linear warmup followed by a constant rate, and batch size 256. There is no universal checkpoint budget or seed count: converged base checkpoints and later matched interventions use experiment-specific budgets. The corresponding experiment descriptions state these budgets and seed counts. The primary causal settings use matched 60k-step checkpoints and five seeds. The coefficient MSE is the primary task and the support BCE is auxiliary (App. G).

E METRICS

Query and operator protocols. *Direct query cosine*: cosine between the predicted and true full code, with no permutation matching or fitted operator. The primary intervention metric is the sparse-set minus dense-set direct-query cosine after forwarding every query independently. Its evaluation set has 128 episodes and eight queries per episode, matching the training layout. *Per-atom operator alignment*: within an episode, fit an effective operator from a sufficiently large query set, form $\boldsymbol{M} = \widehat{\boldsymbol{P}}\boldsymbol{A}^{\natural}$, Hungarian-match rows, and score row-wise $|\cos|$ against one-hot. We retain this diagnostic in the canonical evaluation records and additional mechanism analyses, but not as the primary causal metric. *Dense operator fit*: R^2 of the best single linear map fitted to, and scored on, the same isolated dense-query outputs (in-sample). For these interventions this is a secondary diagnostic estimated on a separate 64-query set, so each output row's k unknowns are fitted from 64 queries. Operator fits from an eight-query set are not interpreted, especially at k=8.

Evaluation sets. Episode all-atom alignment success is reported from two evaluation sets: the standard set ($0.97/0.92/0.56/0.23$) and the matched set shared with the classical estimator for the oracle comparison (Fig. 6: $0.93/0.83/0.53/0.20$). The matched oracle set is small ($\approx$30 episodes per cell, because the same-context claim requires scoring both methods on identical contexts), so the differences are binomial sampling at that size, not model changes. At 30 episodes an observed rate of 1.0 carries a one-sided 95% lower confidence bound of about 0.90, so the classical result is exact recovery on every tested episode rather than a certified population rate. A solver-only measurement on 1000 fresh episodes per k under the training distribution gives recovery on $1000/1000$ at k=$3/5/8$ (one-sided 95% lower bound 0.997) and $995/1000$ at k=12. Each returned k=12 failure has feasibly rescaled volume above the truth, certifying suboptimality of that returned solution. These experiments do not independently determine whether the five original contexts satisfy an identifiability condition. The instance-level scattering certificate above is available only at $k = 3$. All five recover with fresh contexts at n_{ctx}=256 from the same dictionaries, and at that context length the rate is $1000/1000$. Continuous errors accompany the counts: the median aligned dictionary error $\|\tilde{\boldsymbol{A}} - \hat{\boldsymbol{A}}\|_F/\sqrt{k}$ is at most 2×10^{-6} at every k, with maxima $0.21/4 \times 10^{-6}/1 \times 10^{-5}/0.32$ at k=$3/5/8/12$, so successes sit far inside the 0.9 alignment threshold except in the thin tail. The sample-size dependence is consistent with the $k \log k$ intuition. Fig. 6 draws the same-context comparison. Both sets use 64 queries per episode. The episode-exact level is itself protocol-conditional

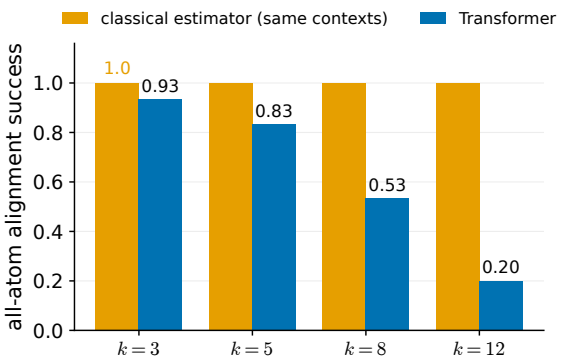

Figure 6: **The context remains classically sufficient.** Episode all-atom alignment success on the matched oracle set at n_{ctx}=128: both methods receive exactly the same 128 context observations per episode, so the comparison holds the available information fixed. The volume-minimization L-ADMM estimator (gold) reaches 1.0 on every matched evaluation episode at every k, exactly the contexts where the Transformer (blue) degrades. The oracle is not given the dictionary or the supports. It solves the same unsupervised problem with unlimited compute per episode. The gap is therefore a property of the trained network and its training regime, not of the information available in the context (≈30 episodes per cell under the matched protocol; the solver-only measurement above covers 1000 episodes per k). Longer *test* context does not rescue the trained network at k=12 ($0.20 \to 0.20$ at n_{ctx}=256), so the gap is not an inference-time context limit either.

(the per-episode operator fit improves with more queries), so only same-shape comparisons are drawn anywhere in this paper.

Relation between the two recovery levels. Query-level direct recovery and episode-level all-atom alignment measure different events with different invariances. The latter row-normalizes $\widehat{\boldsymbol{P}}\boldsymbol{A}^{\natural}$ and is invariant to independent row gains. Full-code query cosine is not. Neither event contains the other (a diagonal operator with unequal positive gains can pass row alignment while failing query cosine). Empirically, the $k = 12$ all-atom rate is 0.20–0.23 versus query recovery 0.545. A query scores its own s-dimensional problem, while the fitted episode operator must align all k atoms. With per-query selection 0.584, errors distributed across atoms explain the observed lower episode rate without making it a logical necessity. Neither level substitutes for the other: the decomposition of Sec. 5 requires a query-level event that can be conditioned on support correctness, while the classical comparison requires an episode-level criterion matched to an estimator that returns an entire dictionary.

Metric definitions. *Query-level:* support exact match ($\widehat{S}$=S from support logits > 0), support F1, coefficient R^2, and the support-head-independent recovery event R used in the decomposition (full-vector cosine of the coefficient output with the true code > 0.9). *Episode-level:* we fit the effective linear operator $\widehat{\boldsymbol{P}}$ from an episode's queries by least squares (n_{query}=64 $\gg k$), form $\boldsymbol{M} = \widehat{\boldsymbol{P}}\boldsymbol{A}^{\natural}$, row-normalize, and score how close $\boldsymbol{M}$ is to a (signed) permutation. *Episode all-atom alignment success* (called episode-exact in historical artifacts) is the fraction with all atoms above 0.9. It certifies directions after matching, not canonical coefficient gains. *Operator-level:* query Jacobian $\boldsymbol{J}$, ambient error $\|\boldsymbol{J} - \boldsymbol{P}^{\natural}\|_F / \|\boldsymbol{P}^{\natural}\|_F$, projected active-subspace error E_{active}, participation and participation ratio. *Hungarian validation:* matching output slots to atoms before scoring raises episode recovery on only $\sim 3\%$ of episodes, all canonicalization near-ties (swapped-pair lead-gap < 0.08). It masks no well-separated error. We keep query-level and episode-level metrics visibly distinct throughout.

Table 3: **Functional identification from the isolated-query Jacobian.** The learned derivative is far from $\boldsymbol{P}^\natural$ in ambient space (relative Frobenius distance over all of $\mathbb{R}^k$), close to the correct restricted inverse on active tangent directions (E_{active}, the only directions sparse data constrain), and has effective dimension near the sparsity $s = 2$ rather than k. Ambient values are means over query-level errors from a 30-episode flagship bank. The E_{active} values are medians over correct-support queries and the PR values are means over all queries on a separate 48-episode bank from the same converged model. A large ambient distance with a small E_{active} identifies a support-dependent rather than global operator; masked versus local requires normal-direction reference distances.

k	$\|\boldsymbol{J} - \boldsymbol{P}^\natural\|_F / \|\boldsymbol{P}^\natural\|_F$	projected E_{active}	$\mathrm{PR}(\boldsymbol{J})$
3	0.65	0.13	1.96
5	0.90	0.12	2.28
8	1.29	0.22	2.69
12	1.17	0.26	2.96

F Functional analyses

F.1 Corrected output-surrogate analysis

On valid sparse queries all three references produce identical outputs (App. B), so fitting the network's *output* cannot identify the operator family. We verify this directly. Fitting ridge surrogates (with intercept) to held-out support-shuffled outputs and scoring active-coordinate R^2, a single context-global linear map (H1), per-atom shared rows with an oracle support mask (H2), and per-support operators (H3) agree to within 0.6% at every k ($k5$ $0.994/0.994/0.995$, $k8$ $0.988/0.989/0.990$, $k12$ $0.966/0.968/0.972$). The residual support-specific advantage is at most 0.005 R^2 ($k12$, 95% CI $[0.003, 0.005]$), an order of magnitude below the contiguous-batch artifact it replaces. An earlier version fit the support-conditioned surrogate on *contiguous same-support* query batches. Because the model attends across all query tokens, same-support batching inflates the apparent support-conditioned advantage ($k8$ H3$-$H1 ≈ 0.05), which collapses under support-shuffled evaluation. We therefore use output surrogates only to demonstrate behavioral non-identifiability, not to identify the mechanism.

Jacobian. $\boldsymbol{J}(\boldsymbol{x}_q) = \partial \hat{\boldsymbol{s}}_q / \partial \boldsymbol{x}_q$ by autograd, validated against central finite differences (isolated single-query token, per-setting mean relative Frobenius discrepancy 2.1×10^{-4}–1.6×10^{-3} for k=3–12, means over up to three queries per episode, raw per-query maxima not retained). Support-overlap distances between Jacobians use Frobenius distance. Effective dimension is the participation ratio of $\boldsymbol{J}$'s singular values, $\mathrm{PR}(\boldsymbol{J}) = \left(\sum_i \sigma_i\right)^2 / \sum_i \sigma_i^2$, which equals r when exactly r singular values are equal and the rest are zero, and interpolates smoothly otherwise.

F.2 Held-out reference-field discrimination

For true support S, define the support-masked and gated-local fields $M_S = E_S E_S^\top \boldsymbol{P}^\natural$ and $G_S = E_S(\boldsymbol{A}_S^\natural)^\dagger$. Let Π_S project onto $\mathrm{col}(\boldsymbol{A}_S^\natural)$ and $Q_S = I - \Pi_S$. The registered normal intervention v_N is the unit top right singular vector of $(M_S - G_S)Q_S$, fixed using only A and S. At the sparse base queries used here the canonical selectors of Sec. 2 return the true support, so the true-support fields evaluated below are the completed maps' local branches. We forward $x \pm \epsilon v_N$ independently at $\epsilon \in \{0.001, 0.003\}$ and compare the central response with $\boldsymbol{P}^\natural v_N, M_S v_N, G_S v_N$. Errors use the common separation $\|(M_S - G_S)v_N\|$. Queries are aggregated within episode before bootstrap inference. Eligible/requested queries are $187/192$, $190/192$, $154/192$, and $128/192$ at $k = 3/5/8/12$; the objective and support-step packages below retain $681/768$ and $1144/1280$ under their own preregistered eligibility rules (intersection-correct and base-correct support, which differ, so only winner signs are compared across packages). Table 5 reports the resulting response errors.

The pooled $k = 5/8/12$ local advantage is 0.523 (episode-bootstrap 95% CI $[0.466, 0.579]$). $k = 3$ is the preregistered near-tie. A second implementation reproduces all directions, errors, intervals, and verdicts to 6.8×10^{-14}.

Table 4: **The complete first-order account on one common query bank.** All quantities on ONE common isolated-query bank per k (32 episodes $\times$ 6 sparse queries, all-query medians; the value in parentheses is the correct-support-conditioned median). This bank differs from the flagship bank of Table 1, so levels differ while the ordering agrees. Tangent $= \|(\boldsymbol{J} - \boldsymbol{P}^{\natural})\boldsymbol{\Pi}_S\|_F$, normal $= \|\boldsymbol{J}(\boldsymbol{I}-\boldsymbol{\Pi}_S)\|_F$, and dist $= \|\boldsymbol{J}-\boldsymbol{G}_S\|_F$ with $\boldsymbol{G}_S = \boldsymbol{E}_S(\boldsymbol{A}_S^{\natural})^{\dagger}$, where $\mathrm{dist}^2 = \mathrm{tangent}^2 + \mathrm{normal}^2$ holds per query before aggregation (verified to 10^{-14}; the displayed columns are independent medians and need not satisfy it); "fl" normalizes by $\|\boldsymbol{G}_S\|_F$ (floor units) and "own" by $\|\boldsymbol{J}\|_F$. $\mathrm{PR} = (\sum_i \sigma_i)^2 / \sum_i \sigma_i^2$. The full tangent block (including inactive output rows) is small for sparse training, the distance to the gated-local reference is dominated by the normal block and is not near zero, and the dense-trained control sits several times farther from the gated-local reference on identical queries: the sparse-trained endpoint is much closer to gated-local than the dense control is, but does not attain that reference. The three-reference ranking is in Table 1.

arm	k	ambient	tangent/fl	normal/fl	dist/fl	dist/own	$\mathrm{PR}(\boldsymbol{J})$	$\mathrm{PR}(\boldsymbol{G}_S)$
sparse	3	0.66	0.15	0.49	0.52	0.51 (0.50)	1.94	1.92
sparse	5	0.87	0.20	0.77	0.85	0.70 (0.66)	2.27	1.96
sparse	8	0.99	0.29	1.22	1.27	0.83 (0.80)	2.68	1.97
sparse	12	1.00	0.51	1.80	1.95	0.96 (0.91)	2.91	1.98
dense	3	0.09	0.09	1.20	1.25	0.79	2.37	1.92
dense	5	0.22	0.18	1.86	1.90	0.92	3.78	1.96
dense	8	0.50	0.39	5.48	5.48	0.99	3.87	1.97
dense	12	0.70	0.56	10.58	10.59	1.00	3.61	1.98

Table 5: **Held-out finite directional responses.** Fresh bank seed 2026090701, correct-support subset fixed before response errors. Lower is closer. Errors are episode means of within-episode query medians; the local advantage is the episode mean of the paired per-episode difference $\min(\text{masked}, \text{global}) - \text{local}$, so it is not the difference of the displayed columns.

k	global	support-masked	gated-local	local advantage
3	1.942	0.937	0.923	0.014
5	1.615	0.821	**0.613**	0.193
8	1.881	0.891	**0.390**	0.501
12	2.343	1.004	**0.130**	0.874

An output-level test agrees. On full-dimensional queries the support-masked account places the output support on the code's top-s coordinates by construction, while the network's top-s output set matches the truth's at rates $0.77/0.54/0.31/0.12$ for $k = 3/5/8/12$. The network's dense outputs are therefore not those of the canonical $\mathrm{TopK}_s(\boldsymbol{P}^{\natural}\boldsymbol{x})$ support-masked extension, independently of any derivative.

The same diagnostic tracks training interventions (Table 6). At a common 60k budget, all three standard sparse/dense $k = 5$ seed pairs reverse from gated-local to global (paired shifts 2.399–2.676) at nearly identical sparse-query cosine. At $k = 8$, all three 60k seeds already reverse and the seed-0 result persists at 200k. Converged-endpoint multi-seed replication remains open.

For the objective contrast, exact historical SageMaker source bundles were retrieved and their active paths produced byte-identical batches, outputs, losses, and five-step updates under both loss settings. A fresh paired experiment then held architecture, data, optimizer, budget, and seed fixed: jointly removing support BCE and off-support ℓ_1 changed gated-local to support-masked, with cosine $0.96282/0.96289$. Under pure MSE, expanding support with matched expected energy changed support-masked to global in all five seeds, with cosine $0.9609/0.9537$. A third preregistered comparison on a fresh bank (seed 2026091001) separates the two support-side terms with existing runs: the arm trained with support BCE removed but off-support ℓ_1 kept (λ_{sup}=0, λ_0=0.01, three seeds) is closest to support-masked in three of three seeds (margin 0.105, seed bootstrap $[0.074, 0.152]$; mean fresh cosine 0.960 against 0.970 for the standard arm), while on the same bank the standard arm stays gated-local (0.139, $[0.062, 0.199]$) and the pure MSE arm stays support-masked (0.114, $[0.091, 0.145]$). All three eligibility populations (all queries, intersection-correct, standard-correct) give the same signs. The objective-axis effect is therefore carried by the support BCE; the fourth cell (BCE kept, ℓ_1 removed) is reported below under *The remaining loss cell*. Every experiment was

Table 6: **Training design selects different references at matched task performance.** Margin is distance to the closest alternative minus distance to the winner. The dense row's registered quantity is the per-seed local-to-global shift quoted in the text. Its winner margins were not a registered gate.

training	closest	seeds	margin (95% CI)	cosine
objective contrast (one matched package)				
standard sparse	gated-local	3/3	.147 [.068, .221]	.96282
pure MSE, $s' = 2$	support-masked	3/3	.131 [.117, .138]	.96289
query-support contrast under pure MSE (a separate matched package)				
pure MSE, $s' = 2$	support-masked	5/5	.114 [.100, .135]	.9609
pure MSE, matched $s' = 3$	global	5/5	.172 [.075, .269]	.9537
independent dense endpoint				
standard dense	global	3/3 ($k = 5$)	–	.962–.977

preregistered on a fresh bank and independently reanalyzed; manifests and denominator accounting are in the project records named by the result manifest.

Table 7: **The two axes at k=8.** Projected family coordinates (α, β) (seed means of query medians) and the nearest reference by seed count, under the paper's historical sampler (five seeds) and a strict $\kappa \leq 5$ sampler (three seeds); sparse cosines are level within 0.03 in every arm. Intersection-correct: every arm decodes the query's support (App. F.2).

sampler	arm	(α, β)	nearest (intersection-correct)	nearest (all queries)
historical	standard	$(0.12, 0.00)$	gated-local 5/5	gated-local 5/5
historical	pure MSE, s'=2	$(0.21, 0.03)$	gated-local 5/5	gated-local 5/5
historical	pure MSE, matched s'=3	$(0.65, 0.30)$	support-masked 5/5	gated-local 3/5
strict	standard	$(0.33, 0.02)$	gated-local 3/3	gated-local 3/3
strict	pure MSE, s'=2	$(0.52, 0.07)$	support-masked 2/3 (near-tie)	gated-local 2/3
strict	pure MSE, matched s'=3	$(0.78, 0.39)$	support-masked 3/3	support-masked 3/3

The two axes at k=8. Under the paper's own sampler the standard arm is the five existing k=8 models and the two pure MSE arms (fixed s'=2; s'=3 with matched expected energy) are new, five seeds each, 60k steps, scored on one fresh bank of 64 episodes with four exact-s=2 queries (seed 2026091006). Fresh sparse cosines are $0.885/0.875/0.873$ (arm means; all fifteen runs in 0.851–0.891). Held-out normal-response errors (seed means, gated-local field units) are $2.03/0.99/0.43$ to global/masked/local for the standard arm, $1.97/0.95/0.49$ for pure MSE s'=2, and $1.46/0.83/0.90$ for the energy-matched s'=3 arm (intersection-correct queries): the objective step leaves gated-local nearest on every population and seed; after the support step support-masked is nearest on all five seeds among intersection-correct queries, while among all queries and among standard-correct queries gated-local remains nearest on seeds 0, 3 and 4 (support-masked on 1 and 2). Family coordinates are $(0.12, 0.00)$, $(0.21, 0.03)$ and $(0.65, 0.30)$ (the standard arm's 0.12 on this bank against 0.15 on the held-out bank of Table 8); the objective step raises α by 0.08 (five of five seeds) and the support step raises β by 0.27 (five of five), with the family explaining $0.07/0.13/0.53$ of the residual. The registered expectation that the k=5 nearest-reference switches would recur was not met for either step; the coordinate movement is the result. *The remaining loss cell.* Keeping the support BCE and removing only the off-support ℓ_1 (k=5, three seeds, one fresh bank of 64 episodes with four exact-s=2 queries, seed 2026091101) leaves gated-local nearest on every seed and every population, with gated-local margins $0.159/0.066/0.129$ on intersection-correct queries and $0.181/0.089/0.179$ on all queries and family coordinates $(0.42, 0.01)$, $(0.48, 0.02)$, $(0.39, 0.02)$; the BCE-removed, ℓ_1-kept arm scored on the same bank is support-masked nearest on every seed and population (margins $0.137/0.110/0.089$ intersection). On this bank the standard arm itself is gated-local nearest on all and standard-correct queries in every seed but by -0.021 on intersection-correct queries in seed 1, a near-tie between the two support-dependent references. *Strict conditioning.* A second k=8 experiment retrains all three arms (standard; pure MSE s'=2; pure MSE s'=3 with matched expected energy) with a sampler that rejects every dictionary above the nominal condition-number threshold of 5 (App. D; realized maximum 4.9999 against 6636 for the same generator seed under the historical capped sampler), three seeds per arm, 60k steps, and scores them on one fresh bank of 64 episodes with four exact-s=2 queries each (seed 2026091004). Fresh sparse

Table 8: **Continuous-family coordinates.** Query medians of the least-squares (α, β), the residual to the fitted family member r_{fam} in gated-local floor units, the fraction of the squared residual to the nearest vertex removed by the fit, and the fraction of held-out normal directions on which the fitted family beats the best vertex. Vertices: gated-local $(0, 0)$, support-masked $(1, 0)$, global $(1, 1)$. The k=3 normal subspace is one-dimensional, so it has no held-out direction.

arm	k	α	β	r_{fam}	residual removed	family beats vertex
standard sparse	3	.49	.03	.31	.33	–
standard sparse	5	.37	.01	.62	.17	.18
standard sparse	8	.15	.00	.98	.09	.39
standard sparse	12	.01	.00	1.65	.02	.41
standard dense	5	1.01	1.02	.16	.22	.48
standard dense	8	.97	1.01	1.38	.18	.39
standard sparse (objective bank)	5	.41	.02	.64	.16	.23
pure MSE, s'=2 (objective bank)	5	.62	.09	.62	.27	.34
pure MSE, s'=2 (support bank)	5	.60	.09	.58	.29	.37
pure MSE, matched s'=3	5	.88	.57	.45	.68	.76

cosines are $0.883/0.862/0.886$ (arm means; per-seed range 0.846–0.894), so the arms are behaviorally level within 0.03. On intersection-correct queries the standard arm's gated-local margin is $0.20/0.24/0.16$ by seed; the pure MSE arm's support-masked margin is $-0.03/0.14/0.03$, a near-tie between the two support-dependent references; and the energy-matched arm's global margin is $-0.24/-0.33/-0.15$, with support-masked nearest. Family coordinates (seed means of query medians) move from $(0.33, 0.02)$ to $(0.52, 0.07)$ to $(0.78, 0.39)$ across the three arms, with the support step raising β by 0.32 ($0.34/0.28/0.35$ by seed) and the family explaining $0.10/0.17/0.55$ of the residual. The registered per-seed behavior threshold (0.85) was missed by one pure MSE seed at 0.846; the arm-level matching criterion is the one we report, and every number above is the measured value for all three seeds. An independent reimplementation reproduces every margin and coordinate to 2×10^{-6}.

A continuous reference family. Nearest-vertex classification does not by itself exclude a continuous partial normal response. For true support S the two-parameter family $\boldsymbol{J}(\alpha, \beta) = G_S + \alpha(M_S - G_S) + \beta(\boldsymbol{P}^{\natural} - M_S)$ contains the three references at $(0, 0)$, $(1, 0)$, $(1, 1)$; both increments annihilate $\boldsymbol{A}_S^{\natural}$, so every member agrees with the references on every sparse query with support S, and their output rows are disjoint (S and S^c), so the least-squares coordinates of a Jacobian are two independent projections. Table 8 fits (α, β) to the exact autograd Jacobian at every correct-support query (fresh bank seed 2026090701; the selection packages on their own banks) and tests the family on a held-out normal direction: the coordinates are fitted from the registered direction v_N alone and then predict the response along the second right singular vector of $(M_S - G_S)Q_S$. For the standard sparse model the fitted point lies near the gated-local vertex, $\beta \approx 0$ at every k and α shrinking from 0.37 at k=5 to 0.01 at k=12, and the family does not explain the residual to that vertex: the fitted family removes $17/9/2\%$ of the squared residual at k=5/8/12, and its prediction beats the best vertex on $18/39/41\%$ of held-out queries, a chance-level rate at the larger k where the residual it removes is already negligible. The remaining distance to gated-local is off-family, consistent with the query-adaptive derivative described in Sec. 3; the nearest vertex is gated-local at every k, and the distance to it grows with k. The coordinates do recover a vertex when a network sits on one: the dense control lands at $(1.01, 1.02)$ at k=5 and $(0.97, 1.01)$ at k=8. On the selection map the objective axis moves α ($0.41 \to 0.62$ in the pooled query median; all three seed pairs) and the query-support axis moves β ($0.09 \to 0.57$; all five seed pairs). The energy-matched s'=3 arm is the one arm that is a genuine interior point of the family: at $(0.88, 0.57)$ the family prediction beats the best vertex on 76% of held-out queries and removes 68% of the squared residual, so its "global" label names the nearest vertex of a partial move from support-masked toward global.

What the ranking measures: a size-matched control. The three references share the tangent action, so their distances to a Jacobian $\boldsymbol{J}$ differ only through its normal block $\boldsymbol{J}Q_S$, and gated-local is the reference whose normal block is zero. To separate the size of the network's normal block from its orientation we build a control from each measured Jacobian: keep its tangent block, replace its normal block by a Gaussian random block on the same subspace rescaled to the same Frobenius norm (twenty draws per query, paired across arms by a shared random key), and rank the control by the Table 1 distance and by the linearized response along v_N. Table 9 shows two things.

Table 9: **A size-matched randomly oriented normal block is nearest gated-local on almost every query; the network's block is oriented toward support-masked at small k.** Standard sparse model on two fresh banks (bank 1 seed 2026091301, bank 2 seed 2026091401; 64 episodes $\times$ 4 queries); control = network tangent block kept, normal block replaced by a random block of the same norm (twenty paired draws per query). Nearest-gated-local share by Table 1 distance / by linearized v_N response; "normal" is the network's median normal-block norm in floor units; $\cos(N, B_1)$ is the median cosine between the normal block and the masked increment $B_1 = M_S - G_S$ (control: median $\pm$ SD over draws); "α beyond draws" is the share of queries whose network α exceeds all twenty draws (chance 5%). Last two columns: the dense-trained network's share nearest global and its control's share nearest gated-local.

		nearest gated-local (%)			$\cos(N, B_1)$		α beyond	dense-trained (%)	
k	bank	network	control	normal	network	control	draws (%)	global	ctrl. local
5	1	67 / 64	97 / 96	0.70	0.49	0.00 ± 0.26	52	91	80
5	2	64 / 65	96 / 95	0.79	0.50	0.00 ± 0.25	47	98	81
8	1	90 / 90	98 / 98	1.05	0.27	0.00 ± 0.15	52	86	91
8	2	90 / 91	99 / 99	1.19	0.23	0.00 ± 0.14	47	89	90
12	1	98 / 97	99 / 99	1.84	0.02	0.00 ± 0.09	20	67	97
12	2	98 / 98	100 / 100	1.61	0.02	0.00 ± 0.09	22	55	98

First, a normal block of the network's size with random orientation is nearest gated-local on almost every query: the control ranks gated-local on 96–100% of queries, as the geometry predicts, since $d^2_{\text{masked}} - d^2_{\text{local}} = \|B_1\|_F^2 - 2\langle N, B_1\rangle$ with $B_1 = M_S - G_S$ and a random N aligns with B_1 only by a rare fluctuation. The ranking is independent of orientation only when $2\|N\|_F \le \|B_1\|_F$; the control establishes the outcome under isotropic orientation, not for every orientation, which is why the second observation is needed. Second, the network's block is not a random block: its cosine with B_1 is $0.49/0.27$ at k=5/8 against $0 \pm 0.26/0.15$ for the control, and its family coordinate α exceeds every one of the twenty paired draws on about half of the queries (chance $1/21$); at k=12 the cosine is 0.02 and the network sits at the vertex. The nearest-gated-local label is therefore fixed by the size of the off-manifold response, while the network's orientation toward support-masked at small k is a separate, measured fact. For the dense-trained network the same construction fails in the other direction: the network is nearest global while its size-matched control stays nearest gated-local, so at the global end the label reports orientation. Two further tests bear on mechanism. On queries whose coefficient readout selects a wrong support (21–103 of 256 per k), the distance from $\boldsymbol{J}$ to the gated-local field on the readout's support minus the distance on the true support has median within ± 0.1 floor units on both banks (sign not stable across banks), so the readout does not organize the derivative. At off-manifold probes where the canonical selectors disagree, the Jacobian is 2–4$\times$ farther from the gated-local field on any candidate support than at the base point. Completed functions tell the same story: on normal perturbations of relative radius 0.3 the gated-local map has the smallest response error on $33/65/97\%$ of probes at k=5/8/12, with winner error $0.6/0.5/0.1$ of the winner-to-runner-up separation; on inputs denser than the training set no candidate is adequate (ratios 0.75–0.87). These analyses are exploratory: no gate was preregistered, and the two banks are the replication unit.

Direction, not size, in the selection map. The same control applied to every training arm asks whether the objective and query-support steps of Sec. 4 change the orientation of the normal block or only its norm. For each step and each paired seed we take the network's change in the target contrast (support-masked minus gated-local distance for the objective step; global minus support-masked for the support step; negative means toward the step's reference) and the same change for the size-matched control built from the same Jacobians with the same random draws. Table 10 reports medians over seeds on two populations, all queries and the intersection on which every arm reads the correct support, together with the network's change in the projected coordinates. The control's contrast does move under the support step, because a larger normal block by itself changes the reference distances, but by a fraction of the network's move, and its projected coordinates stay at zero in expectation. The one seed in which the network's objective-step move under the paper's k=8 sampler is smaller than its control's is visible in the last column. The arm medians plotted in Fig. 4 are the all-query values on these banks: standard $(0.37, 0.02)$, pure MSE $(0.58, 0.12)$,

Table 10: **Training moves the normal block's orientation beyond what its size explains.** Per training step, median over paired seeds of the change in the target contrast (floor units) for the network and for its size-matched control, on the all-query and intersection-correct populations; the network's median change in (α, β); and the number of seeds in which the network's change exceeds its control's (all-query, intersection in parentheses). k=5: standard → pure MSE (three seeds), pure MSE → energy-matched s'=3 (five seeds); k=8 under the paper's sampler (five seeds) and the strict $\kappa \leq 5$ sampler (three seeds).

		all queries		intersection			
setting	step	network	control	network	control	$\Delta\alpha$ / $\Delta\beta$	seeds beyond
k=5	objective	-0.18	-0.06	-0.19	-0.04	$+0.20$ / $+0.09$	3/3 (3/3)
k=5	support	-1.08	-0.12	-1.19	-0.14	$+0.30$ / $+0.48$	5/5 (5/5)
k=8, paper's sampler	objective	-0.16	-0.07	-0.20	-0.09	$+0.07$ / $+0.02$	4/5 (5/5)
k=8, paper's sampler	support	-1.23	-0.24	-1.17	-0.18	$+0.42$ / $+0.27$	5/5 (5/5)
k=8, strict	objective	-0.13	-0.01	-0.14	-0.02	$+0.16$ / $+0.06$	3/3 (3/3)
k=8, strict	support	-0.75	-0.06	-0.92	-0.09	$+0.28$ / $+0.27$	3/3 (3/3)

Table 11: **Scale-sensitive fidelity on the selection-map queries** (per-seed means over eligible queries, ranges across seeds). Support accuracy is read from the coefficient output (top-2 magnitudes); it is 1 by construction for the arm(s) in each package's eligibility rule.

arm	norm. sq. error	gain	tangent error	coef. support acc.
standard sparse	.026–.034	.969–.975	.218–.230	1.00
pure MSE, s'=2 (objective bank)	.036–.040	.944–.964	.209–.231	1.00
pure MSE, s'=2 (support bank)	.019–.048	.937–.976	.185–.221	1.00
pure MSE, matched s'=3	.043–.065	.934–.980	.147–.187	.93–.97

energy-matched s'=3 $(0.88, 0.60)$, dense-trained $(1.00, 1.02)$, and the late, uniform, and withdrawn admixture arms $(0.41, 0.03)$, $(0.42, 0.04)$, $(0.36, 0.02)$ (App. J).

Fidelity at matched cosine. Cosine is invariant to a positive gain, while reference discrimination is computed in the canonical coefficient scale. Table 11 therefore reports, on exactly the eligible queries of Table 6, the coefficient normalized squared error, the fitted gain $\langle \hat{\boldsymbol{s}}, \boldsymbol{s} \rangle / \|\boldsymbol{s}\|^2$, and the full tangent-block error $\|(\boldsymbol{J} - G_S)\boldsymbol{U}_S\|_F / \|G_S \boldsymbol{U}_S\|_F$. The objective contrast is matched on all three (per-seed error differences at most .014; pooled tangent difference $-.004$ with episode-bootstrap interval $[-.025, .018]$). The query-support contrast is matched on gain but not on coefficient error: the s'=3 arm is about .02 worse in normalized squared error on every seed and better in tangent error (pooled $-.039$, interval $[-.057, -.020]$). Restricting each seed to queries whose two arms differ by at most .05 in normalized squared error retains 189–208 of 219–233 queries, and on that subset the nearest-vertex majority still moves gated-local to support-masked $(3/3)$ and support-masked to global $(5/5)$, so the preference change is not produced by the fidelity difference.

Orthogonal spans. The three references are not pairwise distinguishable for every invertible dictionary: if $\mathrm{col}(\boldsymbol{A}^{\natural}_S) \perp \mathrm{col}(\boldsymbol{A}^{\natural}_{S^c})$ then $\boldsymbol{P}^{\natural}[S, :] = (\boldsymbol{A}^{\natural}_S)^{\dagger}$ exactly, so $M_S = G_S$ and the separation $\|(M_S - G_S)Q_S\|_2$ vanishes. The shipped errors divide by this separation with no floor, so we report its census over every requested query: on the Table 5 bank the minimum is $.48/.71/1.72$ at k=5/8/12 (medians $1.43/2.41/7.76$) and .34 and .51 on the two selection-package banks, with no query below .05; at k=3, whose normal subspace is one-dimensional, 2 of 192 queries fall below .05 (minimum .038), which is one more reason the k=3 row is the registered near-tie and is not scored. As a negative control we evaluate the sparse k=5 model on 48 episodes whose dictionaries have exactly orthogonal active and inactive spans (condition number at most 5, seed 2026090904). The model still performs the task (correct-support rate .91, median sparse cosine .999); on all 175 eligible queries the support-masked and gated-local errors coincide exactly, while the global error exceeds the gated-local error by more than .25 on 98% of queries (median gap .86). The diagnostic separates the references exactly when the geometry allows it, and reports a tie otherwise.

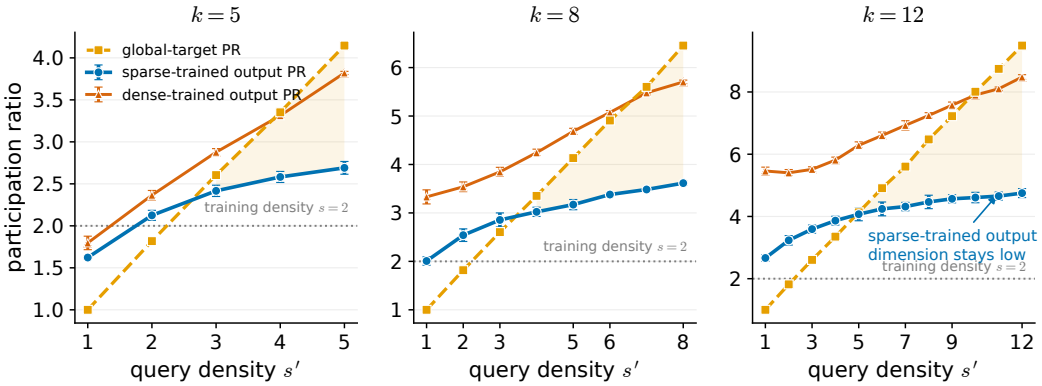

Figure 7: **Fed denser queries than it was trained on, the network answers with sparse-shaped outputs.** We synthesize queries $\boldsymbol{x} = \boldsymbol{A}^{\natural}\boldsymbol{c}$ whose codes have s' active coordinates (the context stays at the training sparsity s=2, dotted line), forward them one at a time, and count how many coordinates the output effectively uses: the participation ratio $\mathrm{PR}(\boldsymbol{v}) = (\sum_i |v_i|)^2 / \sum_i v_i^2$ equals m for m equal-magnitude coordinates and needs no zero threshold, which network outputs would make arbitrary (it is the output-coordinate analog of the singular-value ratio in Table 3). Gold: PR of the true code, which is exactly what $\boldsymbol{P}^{\natural}$ would output (slightly below s' because random magnitudes lower PR, so the target is not idealized). Blue: PR of the network's output (three seeds, SD bars). The references predict differently here: the exact global inverse follows the gold curve, while the support-masked and gated-local references are structurally capped near s. Measured, the sparse-trained network saturates at $2.7/3.6/4.8$ while the target reaches $4.1/6.5/9.5$ (s'=k at k=5/8/12). The relative errors to the global target at s'=k are $0.53/0.78/0.92$, from the same three-seed isolated protocol (seed SD ≤ 0.017). The dense-trained networks of Sec. 4 (orange: five seeds at k=5, 8, the three seeds at k=12) climb with the target and reach $3.8/5.7/8.5$, so the metric does detect tracking when the learned computation is near-global: the saturation of the blue curve is a property of sparse training, not of the measurement. The k=12 dense control is not behaviorally converged at its matched training budget. It demonstrates dimensional tracking (its relative error to the global target remains ≈ 0.56), not an exact inverse. Because all three computations agree on every sparse query (Prop. 1), only off-manifold behavior can separate the global inverse from the support-dependent pair, and this test uses no Jacobian: a derivative-free witness of the same conclusion.

G Training-query distribution intervention

Primary settings. We evaluate pure sparse and pure dense evaluation sets after clearing every training-time mixed/dense-admixture field, assert realized support sizes, and forward queries one at a time. The isolated direct-query cosine gap is 0.113 ± 0.006 at $k5$ (seed range $[0.103, 0.119]$) and 0.259 ± 0.022 at $k8$ ($[0.229, 0.290]$) for the sparse baseline. The dense setting gives -0.006 ± 0.002/-0.016 ± 0.008, and the mixed setting -0.002 ± 0.003/0.009 ± 0.003, at $k5/k8$. Baseline and dense/mixed seed ranges do not overlap. The intermediate fixed-density settings (s'=3, 4 at $k5$ and s'=3, 4, 6 at $k8$) use three matched seeds each, with seed SDs between 0.002 and 0.004. Table 12 reports the absolute per-setting levels behind the primary gaps, so each gap can be checked against the levels: the dense and mixed settings lose only 0.01–0.04 sparse cosine relative to the sparse baseline while closing the dense side entirely, so the endpoint contrast is not produced by one setting failing the easy task.

Dense-fraction sweep. At $k5$, making a training query dense with probability ϵ changes the primary isolated gap $0.113/0.020/0.013/0.003/0.000/-0.006$ for $\epsilon = 0, 0.005, 0.02, 0.10, 0.25, 1$. The endpoints use five seeds and each intermediate ϵ setting uses three matched seeds, with seed SDs of at most 0.004. Thus 0.5% dense supervision removes about 82% of the baseline gap. The fraction is the training-query mixture weight, not a small absolute number: with batch 256, eight queries per episode, and 60k steps, ϵ=0.005 corresponds to roughly 6×10^5 dense queries seen during training. The dose response is strongly nonlinear in the mixture weight: the reduction does not track the share of dense supervision (a proportional reading would give under 1% at ϵ=0.005),

Table 12: Absolute isolated direct-query cosine per training setting (mean ± seed SD, five seeds, 128-episode/8-query pure evaluation sets). The gap column is computed from unrounded per-seed values, so its last digit can differ from the difference of the two displayed columns. The absolute levels rule out one setting failing the easy task: dense and mixed training lose only 0.01–0.04 sparse cosine relative to the sparse setting, so their near-zero gaps come from strong dense performance, not degraded sparse performance.

setting	k	sparse queries	dense queries	gap
sparse s'=2	5	0.961 ± 0.006	0.848 ± 0.007	0.113
mixed	5	0.953 ± 0.009	0.955 ± 0.007	−0.002
dense s'=k	5	0.939 ± 0.006	0.946 ± 0.007	−0.006
sparse s'=2	8	0.895 ± 0.013	0.636 ± 0.019	0.259
mixed	8	0.887 ± 0.012	0.879 ± 0.011	0.009
dense s'=k	8	0.854 ± 0.010	0.870 ± 0.007	−0.016

and no learning theory predicts such a reading, so this describes the response rather than refutes a model. Treating the seed as the replication unit, the derived reductions carry the following uncertainty: the s'=3 reduction is 96% with paired-seed bootstrap 95% interval $[93\%, 101\%]$ and conservative seed-extreme range $[92\%, 101\%]$, and the ϵ=0.005 reduction is 82% with $[80\%, 85\%]$ and $[78\%, 85\%]$. The ϵ=0.005 setting's sparse-side level is 0.951 ± 0.007, one point below the baseline's 0.961 ± 0.006, which is the sense in which the admixture leaves sparse-task accuracy nearly unchanged. On the separate 64-query isolated set, the ϵ=0.005 dense operator fit is already $R^2 = 0.970$. The small remaining direct-query gap prevents a mere-presence claim; the reduction is far from proportional to the mixture weight.

Count versus fraction. Whether the response tracks the share of dense queries or their number is answered by a (ϵ, steps) grid: $\epsilon \in \{0, 10^{-4}, 5\times10^{-4}, 10^{-3}, 2.5\times10^{-3}, 5\times10^{-3}\}$, three seeds each, 60k steps, checkpoints at 15k, 30k and 60k, scored on one fixed isolated bank with the canonical gap statistic; the reduction is $1 - \text{gap}/\text{gap}_0$ paired by seed at the same checkpoint. At 60k the reduction rises monotonically with ϵ: $0.06/0.45/0.64/0.67/0.83$ for 10^{-4} through 5×10^{-3} (per-seed ranges within ±0.06 except 10^{-4}, whose -0.00 to 0.17 is noise around zero). Pairs with the same number of dense queries seen but different fractions give nearly the same reduction: 6×10^4 dense queries as $\epsilon = 5 \times 10^{-4}$ over 60k steps or as $\epsilon = 10^{-3}$ over 30k give 0.45 and 0.37; 1.5×10^5 as 2.5×10^{-3} over 30k or 5×10^{-3} over 15k give 0.49 and 0.52; 3×10^5 as 2.5×10^{-3} over 60k or 5×10^{-3} over 30k give 0.67 and 0.74 (differences -0.08, 0.03, 0.07). Within this design (constant fraction from initialization, 60k steps) the number of dense queries seen orders the response, with roughly 10^4 doing nothing, 6×10^4 removing about 40% of the gap, and 6×10^5 removing 83%, and the three count-matched pairs differ by less than the seed spread; the pairs also differ in training length, so fraction and duration are not separated here. The timing experiments (below, and Sec. 4.3) show that on an already trained network about 3×10^3 dense queries (2,966) delivered over 100 steps remove half of the gap, so the count is not the controlling variable across designs. At 15k steps the two lowest fractions show gaps up to 0.013 larger than the baseline's, differences at the level of the early-training seed spread. Replication of the earlier sweep on this bank: baseline gap 0.117 and $\epsilon = 5 \times 10^{-3}$ reduction 0.83 against 0.113 and 0.82 above.

Removing the support-head BCE. Retraining the four conditions with $\lambda_{\text{sup}} = 0$ (twelve runs: sparse baseline, s'=3, ϵ=0.005, and dense, three seeds each, 60k steps, ordinary encoder, per-seed sparse cosine 0.93–0.96) preserves the response. The sparse baseline stays local with gaps $0.056/0.052/0.055$; s'=3 eliminates the gap ($-0.002/-0.005/+0.003$, indistinguishable from zero and matching the slightly negative dense-reference regime); ϵ=0.005 retains $0.008/0.016/0.014$, a 77% reduction against the λ_{sup}=0 baseline; and dense gaps are $-0.009/-0.001/-0.009$. The BCE therefore amplifies the gap magnitude (joint-loss baseline 0.113 against 0.054) without creating the front-loaded shape (96%/82% against ∼100%/77%). All twelve variants retain λ_0=0.01 on the ground-truth off-support coordinates, so this experiment isolates the support-head BCE rather than every support-aware loss.

Removing all support-side losses and matching energy. Retraining with only the coefficient MSE (λ_{sup}=0 and λ_0=0; five seeds per condition, 60k steps, ordinary encoder) closes that residual.

The rescaling below applies to training only, and all three conditions are evaluated on the same unscaled canonical sparse and dense banks. The $s'=2$ baseline gap is 0.055 ± 0.005 (all five seeds within $[0.049, 0.063]$), and $s'=3$ reduces it to 0.001 ± 0.004, a 98% reduction with paired-seed bootstrap 95% interval $[92\%, 103\%]$. A third condition rescales each $s'=3$ query code by $\sqrt{2/3}$. With zero-mean active coefficients and unit-norm atoms, this matches the baseline's expected target energy and expected query-input energy (realized target-energy ratios to the baseline: 1.51 raw, 1.00 rescaled). Its gap is -0.001 ± 0.002, a 102% $[99\%, 105\%]$ reduction. The baseline and the rescaled condition therefore also face the same expected energy shift from training queries to the dense evaluation bank. Rescaling lowers the minimum active magnitude from 0.3 to about 0.245, and all fifteen runs pass the behavioral gate (sparse cosine 0.94–0.96). A descriptive comparison on the three seeds shared with the $\lambda_{\text{sup}}=0$ runs above finds that removing the off-support ℓ_1 changes the baseline gap by $+0.001 \pm 0.006$. In this comparison the term shows no measurable amplification, and the twofold amplification above is associated with the support-head BCE. The raw and rescaled $s'=3$ conditions are not compared with each other, since their train-to-probe energy shifts differ. Across the raw and rescaled comparisons, the common intervention is the support-size increase, while the explicit support-side losses and expected-energy difference are removed. Proposition 2 and Lemma 2 organize this intervention; the experiment does not supply an optimizer-level account of its selection.

Optimizer control. The first momentum-SGD runs inherited AdamW's numeric weight decay 0.01 while using learning rates 33–333 times larger. Because PyTorch SGD applies coupled L2 regularization, those runs do not isolate the optimizer and their under-convergence is not evidence for a different computation. We therefore train $k = 5$ seeds $0, 1, 2$ with momentum 0.9, learning rate 0.1, and zero weight decay. Each seed independently passes the behavioral gate before locality is examined: final alignment is $0.976/0.993/0.980$, coefficient MSE is $0.046/0.039/0.046$, and all-atom alignment success is $0.750/0.906/0.844$. Their isolated sparse-to-dense gaps are $0.121/0.087/0.130$, giving 0.113 ± 0.018, nearly identical to AdamW's 0.113 ± 0.006 on the same evaluation set. Jacobian participation-ratio medians are $2.355/2.457/2.378$. The sparse local result is therefore not specific to AdamW. This control tests one corrected SGD configuration, not every optimizer or the full response to changes in the training-query distribution.

Fixed-density response. At $k5$, the isolated direct-query gap is $0.113/0.005/-0.006/-0.006$ for training density $s'=2, 3, 4, 5$. At $k8$ it is $0.259/0.066/0.023/-0.005/-0.016$ for $s'=2, 3, 4, 6, 8$. The curve is strongly front-loaded: one density step removes most locality, while $k8$ retains intermediate behavior. The $s'=3$ setting contains no full-dimensional query, so its shift cannot be reduced to training directly on dense inputs. Participation-ratio checks are retained only as secondary diagnostics, since unequal singular values make $\mathrm{PR} = s'$ an invalid reference.

Design and scope. The standard training settings share architecture, sparse contexts, loss, optimizer, and a 60k-step budget. They differ only in the distribution of query codes used in the loss. Because the standard encoder is unmasked and each episode contains eight queries, that intervention also changes what queries can reveal to one another during training. Independent forwarding at evaluation shows that the learned function changed, but cannot by itself rule out those training-time interactions. We address that alternative with the masked training control below.

Evaluation sets. The primary $n_q=8$ set uses 128 episodes and direct full-code cosine. The secondary $n_q=64$ set uses 12 episodes and fits one operator to independently forwarded dense queries. Its dense operator-fit R^2 values for baseline/mixed/dense are $0.825/0.995/0.997$ at $k5$ and $0.675/0.985/0.989$ at $k8$. Seed is the replication unit. Historical trajectory diagnostics followed the dense/mixed settings under the old operator protocol and are not used for timing claims. App. J reports the later matched baseline and intervention trajectories under the canonical direct-query metric.

Operator geometry across training settings. The isolated-Jacobian diagnostics of Sec. 3 track the behavioral gaps across the three settings. On one matched 60k run per arm (96 isolated sparse queries per cell, aggregated over all queries against the true support as means for the ambient error and PR and as medians for E_{active}, so levels differ slightly from the correct-support-conditioned medians of Table 3, which also report a different flagship run), the ambient error $\|\boldsymbol{J} - \boldsymbol{P}^{\natural}\|_F / \|\boldsymbol{P}^{\natural}\|_F$

Table 13: **Changing the distribution of training queries changes the learned function.** Mean ± seed SD over five matched 60k-step runs. The primary metric uses 128 pure evaluation episodes with eight independently forwarded queries, and the dense operator fit uses a separate 64-query set. Baseline and dense/mixed seed ranges do not overlap.

training setting	k	sparse-to-dense cosine drop	dense op-fit R^2
sparse $s' = 2$	5	0.113 ± 0.006	0.825 ± 0.014
sparse $s' = 2$	8	0.259 ± 0.022	0.675 ± 0.016
mixed	5	-0.002 ± 0.003	0.995 ± 0.001
mixed	8	0.009 ± 0.003	0.985 ± 0.003
dense $s' = k$	5	-0.006 ± 0.002	0.997 ± 0.001
dense $s' = k$	8	-0.016 ± 0.008	0.989 ± 0.001

falls from base to mixed to dense as $0.91/0.58/0.22$ at $k = 5$ and $1.13/0.65/0.50$ at $k = 8$, while $\mathrm{PR}(\boldsymbol{J})$ rises as $2.36/3.27/3.80$ and $2.82/3.87/4.36$. The same diagnostics classify the sparse setting as support-dependent and the dense setting as near-global, so this global-versus-support-dependent check is symmetric rather than one-sided. The masked-versus-local comparison is in App. F.2. The projected error E_{active} is small for the sparse arms ($0.19/0.27$ at $k = 5/8$) and, at this budget, larger for the incompletely converged $k = 8$ dense arm (0.49). A 200k resume brings it down ($0.49 \to 0.28$), consistent with Proposition 3's statement that an exact global inverse shares the tangent action, so the residual is approximation error rather than a different tangent block. This comparison is a single run per arm and supplements, rather than replaces, the five-seed behavioral results above.

Query-independence control. We repeat four $k = 5$ conditions with a boolean attention mask: special and context tokens attend only within their prefix, while each query attends to the prefix and itself. No query attends another query, and no context state can read a query. Consequently, a batched forward equals concatenated one-query forwards, and context hidden states are invariant to query values. Five regression tests verify these properties, and measured batched–isolated relative differences are 2.7×10^{-7} to 4.4×10^{-7}. With three 60k-step seeds per condition, the isolated gaps are 0.101 ± 0.004 for sparse training, 0.004 ± 0.005 for density three, 0.019 ± 0.003 for a 0.5% dense admixture, and -0.007 ± 0.001 for dense training. The corresponding standard-encoder gaps are 0.113, 0.005, 0.020, and -0.006. Thus removing information exchange among training queries preserves both the endpoint contrast and the two front-loaded intermediate results.

H Failure, coverage, and the same-context oracle

Exact decomposition. With R the support-head-independent recovery event, $\Pr(R) = \Pr(\widehat{S}{=}S)\Pr(R \mid \widehat{S}{=}S) + \Pr(\widehat{S}{\neq}S)\Pr(R \mid \widehat{S}{\neq}S)$ holds by total probability, and empirically the residual is $< 10^{-3}$. Rates are in Table 14. Because R is defined on the continuous coefficient output rather than a hard support threshold, the large-k selection-dominated conclusion is not a thresholding artifact. The decline in $\Pr(R \mid \widehat{S}{=}S)$ and the rise in median active-coordinate relative error ($0.050 \to 0.321$) show a real decoding term. Deriving support from the coefficient output gives the same ordering on the same isolated bank: the coefficient-derived readout $\widehat{S}_{\mathrm{coef}}$ (the top-$|S|$ coordinates of $|\hat{c}|$) gives $\Pr(\widehat{S}{=}S)/\Pr(R \mid \widehat{S}{=}S)$ of $0.984/0.984$, $0.897/0.966$, $0.829/0.952$, and $0.584/0.851$ at $k = 3/5/8/12$, against the auxiliary-readout values in Table 14, with per-query readout agreement $0.99/0.97/0.94/0.86$. Both decompositions reconstruct the same $\Pr(R)$ exactly. The selection collapse and its ordering are unchanged under the output-tied definition. Oracle masking is an output-level diagnostic. Splitting masked recovery by support correctness on the same bank separates the two populations that the marginal masked rate mixes: masked recovery is $0.991/0.977/0.983/0.971$ on correct-support queries and $0.441/0.598/0.603/0.609$ on wrong-support queries at $k{=}3/5/8/12$. Oracle masking substantially improves wrong-selection queries (at $k = 12$, $0.141 \to 0.609$), but their post-masking recovery remains below the correct-support level (0.971). Removing inactive- coordinate contamination therefore leaves a residual active-coefficient error when selection fails. The marginal masked rate mixes these two populations.

Table 14: **Isolated-query failure decomposition** (2048 queries per k). From k=5 onward, and increasingly with k, support selection is the dominant failure term (at k=3 both terms are small and comparable). Conditional decoding also degrades. Oracle masking removes inactive coefficients and recovers most, not all, of the large-k loss.

k	$\Pr(R)$	$\Pr(\widehat{S}{=}S)$	$\Pr(R \mid \widehat{S}{=}S)$	$\Pr(R \mid \widehat{S}{\neq}S)$	oracle mask
3	0.972	0.983	0.984	0.235	0.982
5	0.886	0.905	0.961	0.170	0.941
8	0.813	0.843	0.933	0.165	0.923
12	0.545	0.584	0.833	0.141	0.820

The decomposition does not by itself identify the computation. Applying the identical decomposition and masking analysis to the constrained query-linear model of App. I on identical evaluation episodes (bank equality asserted; coefficient-derived support readout for both models, since that model's auxiliary head is uninformative) gives, at k=5/8 (two seeds each): $\Pr(\widehat{S}{=}S)$ 0.84–0.89 / 0.77–0.78, $\Pr(R \mid \widehat{S}{=}S)$ 0.95–0.96 / 0.92, $\Pr(R \mid \widehat{S}{\neq}S)$ 0.18–0.25 / 0.14–0.16, and oracle-masking gains 0.07 / 0.16–0.17. The main model's values on the same bank and readout are 0.90/0.83, 0.97/0.95, 0.19/0.13, and 0.05/0.11. An imperfect fixed linear operator therefore produces a similar decomposition pattern: a wrong top-s readout marks queries whose coefficient estimates are poor, whatever the computation. The decomposition locates the failure. The class identification rests on the derivative geometry (App. F), the off-manifold behavior. The control is weaker on the sparse task at k=8 (mean cosine 0.89 vs 0.92), and no k=12 control exists, so the k=12 selection collapse has no linear counterpart.

Classical-reference fairness audit. The reference estimator is the volume-minimization dictionary learning of Hu & Huang (2023b), solved by linearized ADMM (L-ADMM) as implemented here from the published algorithm. It consumes *only* the observation matrix, never any code. Fairness controls:

- **Identical inputs.** Same episodes (seed 909, 30 episodes), same $\boldsymbol{A}^{\natural}$, same context $\boldsymbol{X}_{\mathrm{ctx}}$, same query set, same $n_{\mathrm{ctx}} \in \{32, 64, 128, 256\}$. Hidden codes are unavailable to *both* methods.
- **Initialization.** $\boldsymbol{X}{=}\boldsymbol{I}_k$ after a QR factorization of the context matrix, dual $\boldsymbol{U}{=}\boldsymbol{0}$ (the default specified in that work), with no oracle initialization.
- **Stopping/restarts.** Tolerance 10^{-4} on $\|\boldsymbol{QX}-\boldsymbol{Z}\|_F + \|\boldsymbol{Z}_{\mathrm{old}}-\boldsymbol{Z}\|_F$, max 4000 iterations, a single run per episode (no restarts). A solver exception counts as a failure (codes set to 0), a conservative choice against the classical method.
- **Identical scoring.** We invert the estimated dictionary ($\boldsymbol{P} = \boldsymbol{A}_{\mathrm{est}}^{\dagger}$), predict the query codes, and score with the *same* episode all-atom alignment criterion and canonicalization as the network.

At n_{ctx}=128 the classical estimator recovers exactly for all k while the network caps below, giving the gap between what is recoverable and what the network reaches. At small n and large k this configured estimator also degrades (we cannot separate information insufficiency from solver non-convergence or exceptions in this single-run protocol), which is why we attribute the n=128 gap to the network not reaching a solution that this same estimator does reach, rather than to data insufficiency.

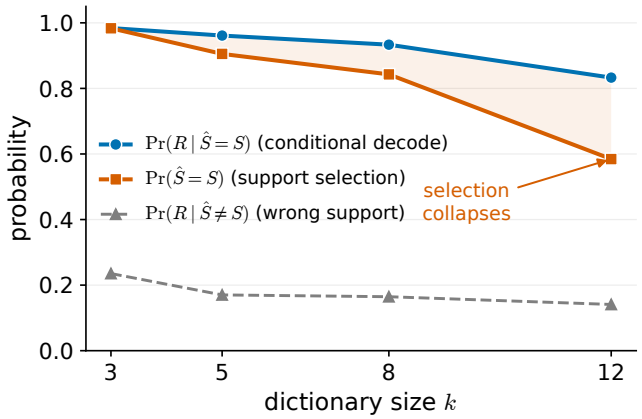

Figure 8: **Large-k failure is selection-dominated.** Exact decomposition of direct recovery on 2048 independently forwarded queries per k. The support-selection factor $\Pr(\widehat{S}{=}S)$ (orange) falls from 0.983 to 0.584 as k grows, while conditional decoding $\Pr(R \mid \widehat{S}{=}S)$ (blue) degrades far less (0.984 to 0.833). Recovery under a wrong support (gray) stays near 0.14–0.24. The gray rate is nonzero because $\widehat{S}$ is read from the auxiliary support head, a diagnostic separate from the coefficient output on which R is defined: the coefficient pathway can decode correctly when that readout misfires near threshold (deriving support from the coefficient output instead gives the same ordering). The decline of the blue curve is a real but secondary decoding term. A separate matched same-context evaluation shows that the classical oracle recovers every tested episode at every k (Fig. 6), and at $k{=}12$, $n{=}128$ every atom is observed. Missing atom occurrences or insufficient context information therefore do not explain the ceiling. The large-k ceiling is dominated by choosing the active set. As the query-linear control below shows, this decomposition locates the failure; the derivative and intervention evidence carry the identification.

I THE VARIATIONAL PROPOSITION: PROOF AND MEASUREMENT

Setting and assumptions. $\boldsymbol{A}^\natural \in \mathbb{R}^{k\times k}$ invertible (the complete case). Queries are drawn as $\boldsymbol{x} = \boldsymbol{A}_S^\natural \boldsymbol{c}$ with $|S| = s$ and $\boldsymbol{c}$ drawn from a density on $\mathbb{R}^s$. Call the resulting distribution μ. Because $\boldsymbol{c}$ has a density, μ-a.e. $\boldsymbol{x}$ has all s coefficients nonzero, hence a unique support, and avoids the pairwise intersections of the pieces (each of dimension $< s$, so μ-null). The target is well defined without any spark condition: $\boldsymbol{x} = \boldsymbol{A}_S^\natural \boldsymbol{c} = \boldsymbol{A}^\natural(\boldsymbol{E}_S \boldsymbol{c})$ gives $T(\boldsymbol{x}) = \boldsymbol{E}_S \boldsymbol{c} = \boldsymbol{P}^\natural \boldsymbol{x}$ by invertibility, so $T = \boldsymbol{P}^\natural|_{M_s}$. The class $\mathcal{F}$ consists of functions $f : \mathbb{R}^k \to \mathbb{R}^k$ with $f = T$ on M_s that are differentiable at μ-a.e. point. The trained network satisfies the differentiability requirement at the measured queries (its Jacobian matches finite differences at mean relative discrepancy $\leq 1.6\times10^{-3}$, App. F). Proposition 3 attains its lower bound in the stated almost-everywhere differentiable exact-interpolant class. The trained smooth network is evaluated as a near-interpolant and is not asserted to attain that field exactly.

Proposition 3 (Unique minimum-Frobenius-norm Jacobian on sparse queries)**.** *Let $1 \leq s \leq k$. Let the query distribution choose a support S of size s and then draw coefficients from a density on $\mathbb{R}^s$. Let $\mathcal{F}$ contain functions $f : \mathbb{R}^k \to \mathbb{R}^k$ that agree with the sparse-code target on M_s and are differentiable almost everywhere under this distribution. For every support assigned positive probability and almost every resulting query $\boldsymbol{x} = \boldsymbol{A}_S^\natural \boldsymbol{c}$,*

$$\boldsymbol{J}_f(\boldsymbol{x})\boldsymbol{\Pi}_S = \boldsymbol{E}_S(\boldsymbol{A}_S^\natural)^\dagger, \qquad \|\boldsymbol{J}_f(\boldsymbol{x})\|_F^2 \geq \|(\boldsymbol{A}_S^\natural)^\dagger\|_F^2. \tag{9}$$

Equality holds exactly when $\boldsymbol{J}_f(\boldsymbol{x}) = \boldsymbol{E}_S(\boldsymbol{A}_S^\natural)^\dagger$. This field is attained almost everywhere by a nearest-subspace extension of the support-gated decoder. The global inverse pays the strict excess

$$\|\boldsymbol{P}^\natural\|_F^2 - \|(\boldsymbol{A}_S^\natural)^\dagger\|_F^2 = \|\boldsymbol{P}^\natural(\boldsymbol{I} - \boldsymbol{\Pi}_S)\|_F^2 > 0 \tag{10}$$

whenever $s < k$.

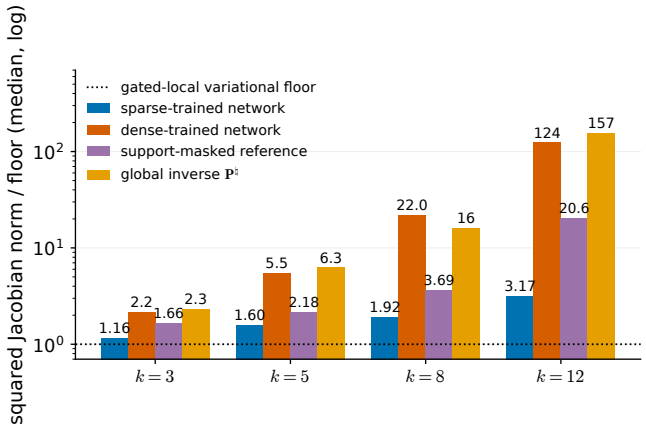

Figure 9: **The Jacobian-norm ordering spans all three references.** Median squared Jacobian norm relative to the gated-local variational floor (dotted line at 1), log scale. The analytic support-masked reference lies between gated-local and global: $1.66/2.18/3.69/20.58\times$ the floor at k=3/5/8/12, against global $2.3/6.3/16/157\times$. The sparse-trained network stays within 1.16–$3.17\times$ of the floor; dense training moves toward the global scale. Sparse, masked, and global bars use correct-support medians; dense bars use all-query medians because its auxiliary support readout is not comparable. The underconverged k=12 dense control is directional only.

Proof of Proposition 3. *(i) The constraint pins the tangential block.* Fix μ-a.e. $\boldsymbol{x}$ with unique support S, and let $f \in \mathcal{F}$ be differentiable at $\boldsymbol{x}$. The whole subspace $\mathrm{col}(\boldsymbol{A}_S^\natural)$ lies in M_s, and on it $f = T = \boldsymbol{P}^\natural$ (a linear map). Hence for every $\boldsymbol{v} \in \mathrm{col}(\boldsymbol{A}_S^\natural)$ the directional derivative is $D_{\boldsymbol{v}} f(\boldsymbol{x}) = \boldsymbol{P}^\natural \boldsymbol{v}$, i.e. $\boldsymbol{J}_f(\boldsymbol{x})\boldsymbol{\Pi}_S = \boldsymbol{P}^\natural \boldsymbol{\Pi}_S$. Also $\boldsymbol{P}^\natural \boldsymbol{\Pi}_S = \boldsymbol{E}_S(\boldsymbol{A}_S^\natural)^\dagger$: on $\mathrm{col}(\boldsymbol{A}_S^\natural)$ both sides map $\boldsymbol{A}_S^\natural \boldsymbol{u} \mapsto \boldsymbol{E}_S \boldsymbol{u}$, and both kill $\mathrm{col}(\boldsymbol{A}_S^\natural)^\perp$, using that the row space of $(\boldsymbol{A}_S^\natural)^\dagger$ is $\mathrm{col}(\boldsymbol{A}_S^\natural)$. *(ii) Pointwise lower bound.* By orthogonality of $\boldsymbol{\Pi}_S$ and $\boldsymbol{I} - \boldsymbol{\Pi}_S$, $\|\boldsymbol{J}\|_F^2 = \|\boldsymbol{J}\boldsymbol{\Pi}_S\|_F^2 + \|\boldsymbol{J}(\boldsymbol{I} - \boldsymbol{\Pi}_S)\|_F^2 \geq \|\boldsymbol{E}_S(\boldsymbol{A}_S^\natural)^\dagger\|_F^2 = \|(\boldsymbol{A}_S^\natural)^\dagger\|_F^2$, with equality iff $\boldsymbol{J}(\boldsymbol{I} - \boldsymbol{\Pi}_S) = 0$, i.e. $\boldsymbol{J} = \boldsymbol{E}_S(\boldsymbol{A}_S^\natural)^\dagger$ (which already annihilates the normal directions). *(iii) Attainment.* The bound requires an f defined on a full neighborhood of M_s. Assign to each $\boldsymbol{x} \in \mathbb{R}^k$ a nearest subspace $S^*(\boldsymbol{x}) = \arg\min_S \mathrm{dist}(\boldsymbol{x}, \mathrm{col}(\boldsymbol{A}_S^\natural))$, with a fixed deterministic rule for ties, and set $\tilde{f}(\boldsymbol{x}) = \boldsymbol{E}_{S^*(\boldsymbol{x})}(\boldsymbol{A}_{S^*(\boldsymbol{x})}^\natural)^\dagger \boldsymbol{x}$. For any $\boldsymbol{x} \in M_s$, every support attaining zero distance returns the same canonical code $\boldsymbol{P}^\natural \boldsymbol{x}$ by Proposition 1. Hence $\tilde{f} = T$ on all of M_s. Every point of M_s with exactly s nonzero coefficients belongs to only one of these subspaces and therefore has a neighborhood on which S^* is constant. Such points have full μ-measure, so $\tilde{f} \in \mathcal{F}$ and it meets the bound μ-almost everywhere. *(iv) Strict excess of the global inverse.* $\|\boldsymbol{P}^\natural\|_F^2 = \|\boldsymbol{P}^\natural \boldsymbol{\Pi}_S\|_F^2 + \|\boldsymbol{P}^\natural(\boldsymbol{I} - \boldsymbol{\Pi}_S)\|_F^2$. For $s < k$, $\boldsymbol{I} - \boldsymbol{\Pi}_S \neq 0$ and $\boldsymbol{P}^\natural$ is invertible, so the second term is strictly positive. □

Remarks. (1) Uniqueness holds at the level of the Jacobian field on M_s. The functional does not constrain f outside M_s, which matches the fact that our measurements probe the operator on or near this set. (2) The statement is a variational characterization of exact interpolants. The network is a near-interpolant, and we verify closeness to the floor directly below rather than assert it. (3) For training density s' the same argument pins a rank-s' tangential block, and at s'=k the pieces are full-dimensional, forcing $f = \boldsymbol{P}^\natural$ wherever differentiable. This is the dense setting of Sec. 4. (4) The bound is attained in the almost-everywhere-differentiable class, not by a continuously differentiable function. Every dictionary line $\{t\, a_i\}$ lies in M_s for $s \geq 1$, so an interpolant differentiable at the origin satisfies $\boldsymbol{J}_f(0)a_i = e_i$ for every i, that is $\boldsymbol{J}_f(0) = \boldsymbol{P}^\natural$, and if $\boldsymbol{J}_f$ is continuous it tends to $\boldsymbol{P}^\natural$ along every sparse line approaching the origin, where the gated-local field differs from $\boldsymbol{P}^\natural$ for $s < k$. A smooth network therefore cannot realize the gated-local field near the intersections of the pieces. The minimum active magnitude of the training codes (App. D) keeps queries away from those intersections, and the measurements of App. F.2 are made there.

Scope: why no optimizer-level selection theorem. We do not attempt one. A rigorous account of which nonlinear extension a trained six-layer Transformer selects would need a theory of its feature-learning dynamics that we do not have. One common route to a tractable training theory is a linearized regime in which the network stays close to its initialization. Such an analysis would address a different regime from the substantial attention reorganization measured here (App. J) and would not by itself explain that feature-learning process. Increasing density raises the minimum interaction order of an exact nonlinear deviation (the lemma below). This is consistent with a front-loaded transition, but it does not prove why training selects one interaction order over another. Results relating SGD stability to input-Sobolev seminorms provide one analogy rather than a premise (Ma & Ying, 2021). Parameter-space implicit regularization and function-space norm characterizations provide broader context (Barrett & Dherin, 2021; Ongie et al., 2020).

Interaction order of a nonlinear deviation.

Lemma 2 (A deviation requires $(s{+}1)$-coordinate interactions)**.** *Write $\boldsymbol{z} = \boldsymbol{P}^{\natural}\boldsymbol{x}$ for the code of an observation and let $\boldsymbol{h}(\boldsymbol{z}) = f(\boldsymbol{A}^{\natural}\boldsymbol{z}) - \boldsymbol{z}$ be a network's deviation from the global inverse. If $\boldsymbol{h}$ is real-analytic near the origin and vanishes on every s-sparse code ($\boldsymbol{h}(\boldsymbol{z}) = 0$ whenever $\|\boldsymbol{z}\|_0 \le s$), then every monomial in its Taylor expansion involves at least $s{+}1$ distinct coordinates.*

Proof. Restrict $\boldsymbol{h}$ to any coordinate subspace of dimension s. The restriction is identically zero, so every Taylor coefficient of a monomial supported inside that subspace vanishes. Any monomial involving at most s distinct coordinates is supported inside some such subspace, hence has zero coefficient. □ □

(For a fixed context, the coefficient map of the reported GELU models (Hendrycks & Gimpel, 2016) is built from affine maps, matrix products and residual additions, softmax, GELU, and layer normalization with positive numerical stabilization. The reported configurations use zero dropout, and dropout is disabled at evaluation. These operations are real-analytic in the query, so the lemma applies to exact interpolants represented by the reported architecture. The trained network is a near-interpolant.) The lemma characterizes the density response: an exact deviation from the global inverse at training density s' requires monomials involving $s'{+}1$ coordinates (third-order interactions at $s'{=}2$, fourth-order at $s'{=}3$, and so on), so each density step raises the interaction order the architecture must realize and maintain across a combinatorially growing family of subspaces. This is an interpretation of the required interaction order, not a theorem about what the optimizer selects.

Measurement. For each converged model and isolated correct-support queries, we compare the per-query cost $\|\boldsymbol{J}\|_F^2$ to the floor $\|(\boldsymbol{A}_S^{\natural})^{\dagger}\|_F^2$ and to the global cost $\|\boldsymbol{P}^{\natural}\|_F^2$ (medians with IQR because the ratio is heavy-tailed through near-degenerate active submatrices, as in App. F): net/floor $= 1.16/1.60/1.92/3.17$ and global/floor $= 2.3/6.3/16/157$ for $k{=}3/5/8/12$. A model-free reference computed from the dictionary distribution alone (fresh draws from the training sampler, with the floor taken as the all-support mean at training density s') gives the per-density version of this comparison at $k{=}5$: the global cost is $5.1/2.7/1.6/1.0$ times the density-s' floor for $s'{=}2/3/4/5$, and the $s'{=}3$ value is the $2.7\times$ ratio quoted in Sec. 4. Because it averages ratios over draws while the values above are correct-support evaluation-set medians, the two $s'{=}2$ values differ. The trained operator is much closer to the variational minimum than the global inverse, but its own gap grows with k: the network/floor ratio reaches 3.17 at $k = 12$. We therefore use this measurement to characterize the selected endpoint, not to claim exact minimization or derive an optimizer preference.

Tangent–normal decomposition (dual normalization). The discriminative first-order quantity is the normal-direction derivative: any accurate interpolant matches the tangential block, so the tangential error is a consistency check while $E_{\perp} = \|\boldsymbol{J}(\boldsymbol{I} - \boldsymbol{\Pi}_S)\|_F / \|\boldsymbol{J}\|_F$ separates the computations. Median $E_{\perp}$ (3 seeds, isolated queries): sparse-trained $0.642/0.774$ vs. mixed $0.807/0.975$, dense $0.911/0.992$, and $\boldsymbol{P}^{\natural}$ $0.907/0.988$ at $k{=}5/8$. Two normalizations must be read together: as a fraction of its *own* (small) norm the sparse-trained network's normal component is not negligible (0.64–0.77: near the floor, not at it), while in absolute units of the floor squared norm its normal component is $\approx$ 0.6–$0.9\times$ the floor versus $\approx$ $15\times$ for $\boldsymbol{P}^{\natural}$ at $k{=}8$, consistent with the cost ratios above. The settings separate cleanly in the correct order under either normalization.

Constrained query-linear model (forward test of the lemma). The context encoder reads only the context tokens (query tokens excluded from the sequence) and emits $\widehat{\boldsymbol{P}}(\boldsymbol{X}_{\mathrm{ctx}}) \in \mathbb{R}^{k\times k}$, and the prediction is forced to $\hat{\boldsymbol{s}} = \widehat{\boldsymbol{P}}\boldsymbol{x}_q$, so the model is query-linear by construction and Lemma 1 makes $\boldsymbol{P}^\natural$ its unique exact solution. Whether training reaches it is an empirical question. At k=5 (two seeds), $\|\widehat{\boldsymbol{P}}\boldsymbol{A}^\natural - \boldsymbol{I}\|/\sqrt{k} = 0.10/0.11$, and sparse and dense alignment coincide within each seed ($0.967/0.963$). At k=8, extending training from 60k to 150k reduces this operator error from $0.436/0.528$ to $0.359/0.261$. Final alignment reaches $0.944/0.952$, and coefficient R^2 reaches $0.798/0.869$. Both seeds move toward the unique linear solution but remain incomplete after 2.5 times the matched budget. Because one operator serves both query families, this model's sparse-to-dense gap cannot reflect a computation switch; an approximate operator need not have exactly zero gap, so we do not read the gap as convergence evidence. The inherited linear support head also cannot classify signed nonzero coefficients: its BCE remains at the Bernoulli-entropy floor 0.5623, while its gradient through the coefficients is only 0.8–1.2% of the coefficient-MSE gradient. We therefore assess this model using coefficient fit and direct operator error, and treat it as supporting evidence because its architecture differs from the main model.

J FORMATION DYNAMICS

Along dense checkpoints (eval every 25, save every 100 steps), t_{50} (steps to reach 50% of the converged value) orders as support representation < local-operator fidelity < recovery at both $k = 5$ and $k = 8$. Measuring local fidelity on oracle support gives the same ordering and a similar transition time to the correct-support diagnostic (de-biased).

When the choice is set. Re-evaluating the masked intervention arms of App. G along their retained checkpoints (every 5,000 steps, three seeds per arm, $k = 5$; the 60k endpoint reproduces the recorded per-seed gaps exactly) shows that under a fixed training distribution the extension forms early and does not migrate later. The sparse baseline's gap forms by step 5,000 (0.065 seed mean) and stays at 0.074 or above from step 10,000 on. The s'=3 setting never forms a substantial support-dependent gap: its gap is $-0.000/-0.004$ at steps 5k/10k and never exceeds 0.011. The 0.5% admixture arm forms a partial early gap (0.051 at 5k) that erodes monotonically to $0.038/0.023/0.019$ at 10k/30k/60k. The largest 30k→60k decrease across arms and seeds is 0.013, so no arm shows a late migration. These trajectories are measured on the query-independence-mask variants at 5k resolution. The ordinary-encoder endpoints match theirs at 60k.

Timing of the admixture. Whether an early-formed extension can still be converted is tested by moving the dense admixture in time. Two k=5 arms receive dense queries drawn from a separate generator at fraction 0.015 inside a window of 20k steps, so that both see the same expected number of dense queries as the 0.5% admixture over 60k (6×10^5; the early and late arms' realized counts are equal per seed to the query): steps 2,001–22,000 (early) or 40,001–60,000 (late); all other batches are the baseline's, and the late arm's parameters equal the baseline's at 40k exactly. Isolated gaps at 60k (three seeds): baseline $0.114/0.119/0.117$, early $0.110/0.116/0.099$, late $0.011/0.018/0.015$, uniform 0.5% admixture $0.023/0.018/0.019$. The early arm reaches 0.019 at 20k, 0.017 at 22k, and returns to 0.058 at 25k, 0.100 at 40k and 0.108 at 60k. Along 100-step checkpoints of seed 0 the late arm's gap falls from 0.112 at 40k to 0.059 at 40,100 (2,966 dense queries), 0.034 at 40,400 and 0.017 at 45k; along 250-step checkpoints the early arm's gap after its window rises from 0.017 at 22k to 0.055 at 25k and 0.089 at 32k. In both directions (same seed) the family coordinates of the sparse-query Jacobian stay within $\alpha \in [0.36, 0.44]$, $\beta \in [0.02, 0.06]$ with the off-family residual in $[0.68, 0.81]$; the off-manifold part of that Jacobian, $\|\mathbf{JQ}\|_F$, falls by about a tenth during the return ($1.43 \to 1.28$, tracking the gap at correlation -0.83 over 23 checkpoints) and rises slightly during the onset. What the admixture converts, and what returns, is the extension to dense queries together with a small normal-direction change of the local map, not its position in the projected family. The resumed runs reproduce the original checkpoints' parameters, optimizer state and generator states exactly at every shared step.

The full Jacobian, paired on identical queries. Coordinate stability does not by itself bound how much the Jacobian moves, so we measure the paired change $\mathbf{J}_t(x) - \mathbf{J}_{\mathrm{src}}(x)$ on the 256 queries of the same bank, in gated-local floor units, split into its tangent and normal blocks and into the part the two-parameter family can see. The undosed continuation of the same 40k state moves its

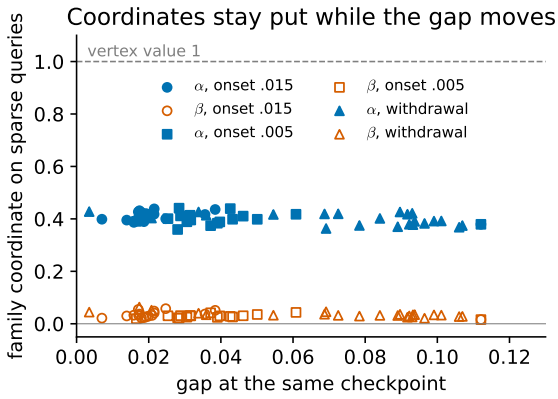

Figure 10: **The projected coordinates stay put while the gap moves.** At the 100-step checkpoints of the onset (ϵ=0.015 and 0.005) and withdrawal arms of Fig. 5, the sparse-query Jacobian's projected family coordinates (α toward support-masked, filled; β toward global, open; vertex value 1) stay within $\alpha \in [0.36, 0.44]$ and $\beta \in [0.02, 0.06]$ while the sparse-to-dense gap at the same checkpoint ranges from 0.004 to 0.112 (57 checkpoints, seed 0).

Jacobian by 0.58–0.69 (query medians, every checkpoint from 40,500 to 48,000), about two thirds of the source's residual to gated-local (0.91), with 0.22–0.25 in the tangent block and 0.52–0.62 in the normal block; two checkpoints of that run 500 steps apart differ by the same amount (0.59–0.63 full, 0.22 tangent, 0.51–0.55 normal), so this is motion, not accumulated displacement. The normal block, which sparse data do not pin (Sec. 2.3), wanders under training while the tangent block barely moves, and the family coordinates stay put because the motion lies in directions the family does not see. A dense-trained network, whose normal block dense data do pin, moves that block by 0.13–0.16 of its size between checkpoints 5,000 steps apart (orientation cosine 0.99; three seeds, four pairs), against 0.44–0.71 for sparse-trained networks under either encoder (orientation cosine 0.80–0.91), so the motion is consistent with motion of the unpinned block and is not explained by the optimizer alone. Under a fixed design the population coordinates fluctuate far less than the design steps move them: over checkpoints 30k–60k of three ordinary-encoder sparse seeds, α has mean 0.38 and SD 0.03 (range 0.34–0.45) and β mean 0.02 and SD 0.01, against design moves of 0.20 and 0.30 in α and 0.48 in β (Table 10). The dose adds little to it: paired per query at matched steps, the dosed continuation's change exceeds the undosed one by 0.01–0.06 at every checkpoint (dosed 0.63–0.74) while the gap falls from 0.11 to 0.01–0.04; the withdrawal arm moves by 0.69–0.79 from its dose edge while the gap returns. At the 60k endpoints (three seeds) the converted and unconverted networks differ by 0.48–0.55 (tangent 0.15–0.19, normal 0.45–0.51) and the returned and unconverted ones by 0.55–0.59, both below the 500-step self-difference of one undosed run. The reference ranking of Sec. 3 and the selection map of Sec. 4 are therefore statements about the population, not about a fixed per-query object: the normal block's size, which sets the ranking, stays at 0.70–0.88 floor units across arms and seeds, and the query-median coordinates stay within $\alpha \in [0.36, 0.44]$ along 57 checkpoints of a fixed design (Fig. 10). Design frozen before computation, independent finite-difference recomputation, and per-query records are in the project records named by the result manifest.

Endpoints across seeds. On the 60k checkpoints of all four arms (three seeds each), the isolated gaps above differ severalfold between the baseline and the late or uniform admixture, while the sparse-query Jacobian's projected coordinates on a fresh 64×4 bank stay close: β is 0.02–0.04 in every arm and seed (paired changes at most 0.02, no move toward global), and α shifts toward support-masked by 0.01–0.12 (baseline 0.29–0.41, late 0.39–0.43, uniform 0.41–0.42, early 0.35–0.38); the largest paired shift equals the baseline's 0.12 cross-seed range. Normal-block norms (0.70–0.88 floor units) and nearest-reference shares remain in overlapping ranges across arms. The dissociation is directional: the dense-query extension moves toward global while the local projection does not.

Share versus number on a trained network. From the same 40k state, admixtures at fractions 0, 10^{-4}, 10^{-3}, 5×10^{-3} and 1.5×10^{-2} for 8k steps (one seed, 100-step checkpoints) give gaps at 45k–48k of 0.113/0.105/0.063/0.031/0.018 (medians; the undosed continuation drifts within 0.098–0.129).

Read against the number of dense queries seen rather than the step, the dosed curves nearly coincide: at about 10^4 dense queries the gap is $0.057/0.050/0.046$ for the three larger fractions, reached after $4{,}900/1{,}000/300$ steps, and at about 3×10^3 it is $0.078/0.067/0.059$. Within a trained network the number seen orders the response and a larger fraction adds a small further reduction at equal number; across designs the exchange rate differs, since 6×10^4 queries from initialization and 3×10^3 on the trained network remove a similar share of the gap. Removing weight decay at the moment the early admixture ends leaves the return unchanged (paired gap differences within 0.017 at every checkpoint while the parameter norm diverges by 6), so the return is not carried by the explicit decay term.

K GENERALIZATION: MODEL WIDTH, SPARSITY, AND COEFFICIENT DISTRIBUTION

Second model width (d_{model}=128). We retrain at half width (d_{ff}=512, same 6 layers / 8 heads and fixed-cardinality sparsity s=2), $k \in \{5, 8\}$, 2 seeds, and 100k steps. The models converge non-floored ($k5$ recovery $0.81/0.91$ and $k8$ $0.59/0.50$, against the $d256$ final-checkpoint references $0.844/0.625$ under the same in-training evaluation; the standard-set protocol of App. E gives different levels). The isolated operator geometry replicates: projected E_{active} remains below ambient error, participation ratio remains near the sparse dimension, and failure is selection-dominated. Table 15 summarizes robustness across the main, second-width, s=3, and coefficient-law settings.

Table 15: Combined robustness. The support-dependent functional structure, operator geometry, and selection-dominated failure hold across a second width, a denser s=3 setting, and three non-Gaussian coefficient laws. These diagnostics separate the support-dependent regime from global; the masked-versus-local comparison is in App. F.2.

setting	E_{active} <amb. (PR$\approx s$)	gap > 0	sel.-dominated
$d256, s2$ (main)	Yes	Yes	Yes
$d128, s2$	Yes	Yes	Yes
$d256, s3$	Yes	Yes	Yes
$d256, s2$, three non-Gaussian coefficient laws	Yes	Yes	–

Sparsity ($s = 3$). We repeat the core battery on already-trained exact-s=3 models ($k \in \{6, 8\}$, same d_{model}=256/6L/8H and objective, no new training). We use fixed-cardinality sparsity to avoid the Bernoulli–Gaussian empty-code confound and validation/test-split head selection. Query-level isolated geometry and support-F1 are primary because episode all-atom alignment is an all-k event that floors at s=3: projected E_{active} stays below ambient error and the participation ratio tracks the sparsity, near s=3.

Coefficient distribution (k=5). We retrain the sparse k=5 condition changing only the distribution of the nonzero coefficients, keeping the minimum-magnitude convention $|c_i| \geq 0.3$. Explicitly, with ε an independent uniform sign and $G \sim \mathcal{N}(0, 1)$: the Gaussian baseline resamples G until $|G| \geq 0.3$ (a truncated Gaussian); the Rademacher-signed folded-Gaussian law is $\varepsilon\,(0.3 + |G|)$ (a shifted folded Gaussian, a distinct law from the truncated baseline); the signed uniform law is $\varepsilon\, U[0.3, 1.3]$; and the signed exponential-tailed law is $\varepsilon\,(0.3 + \text{Exp}(1))$ (a fixed-support adaptation of the Laplace prior); three seeds each. All nine runs pass the behavioral convergence gates, and evaluation sets follow each setting's training coefficient distribution. The isolated sparse-to-dense gap remains positive under every distribution (0.116 ± 0.008, 0.182 ± 0.021, and 0.081 ± 0.010, versus 0.113 ± 0.006 for the Gaussian distribution), and the isolated-Jacobian participation-ratio medians stay between 2.28 and 2.42, near the sparse dimension. Raw gap sizes are not directly comparable across distributions, because how well a dense code can be approximated by two coordinates depends on the distribution: the oracle reference $\mathbb{E}[\max_{\|\boldsymbol{z}\|_0 \leq 2} \cos(\boldsymbol{z}, \boldsymbol{c})]$ is $0.851/0.851/0.797/0.887$ for the Gaussian/Rademacher-signed folded-Gaussian/uniform/exponential-tailed distributions, and the measured dense-side cosine ($0.848/0.828/0.773/0.868$) follows the same ordering at 97–100% of this reference, with the two largest output coordinates carrying 95–97% of the output ℓ_2 norm in every setting. Because the coefficient head is not hard-gated, this reference is descriptive rather than a bound, and we do not claim distribution invariance. The supported statement is that the support-dependent computation persists under bounded, Gaussian-tailed, and exponential-tailed coefficient

distributions, with a distribution-dependent gap magnitude. These behavioral diagnostics do not separate the masked and local references, which is measured in App. F.2.

L CLAIM LEDGER

Table 16 lists every load-bearing statement of the paper with its evidential status: *proved* (an identity or proposition with a complete proof), *measured* (a reference-relative or descriptive quantity on frozen banks, conditioned as stated), *causal contrast* (paired training runs differing in one registered factor), or *open*. Conditioning and seed counts are part of each row.

Table 16: **Claim ledger.** Status vocabulary as defined in App. L.

statement	status	evidence
Global, support-masked, and gated-local maps agree on every sparse query (Prop. 1)	proved	App. B
The global inverse is the unique query-linear extension (Lemma 1); zero risk constrains only the training support (Prop. 2)	proved	App. C
Squared $(s{+}1)$-monomials match the global inverse in value and Jacobian on M_s; identification is class- and resolution-relative	proved	App. C
Gated-local is the unique minimum-Frobenius-norm exact interpolant Jacobian (Prop. 3); deviations need $(s{+}1)$-order interactions (Lemma 2)	proved (exact-interpolant class)	App. I
Standard sparse model is nearest gated-local for $k \geq 5$; the residual to that reference grows with k and is mostly off the two-parameter family	measured (correct-support conditioned; nearest-reference ranking, not a fit)	Tables 1, 4, 5, 8
The nearest-gated-local label is fixed by the size of the normal block (a size-matched random block ranks the same on two banks, $k{=}5/8/12$, five $k{=}8$ seeds); the network's block is in addition oriented toward support-masked at $k{=}5/8$ (cosine with B_1 $0.49/0.27$ vs $0 \pm 0.26/0.15$; α beyond all twenty draws on about half the queries) and not at $k{=}12$; the readout does not organize the derivative; for dense-trained models the same control fails, so the ranking reports direction there	measured (exploratory; two fresh banks)	Table 9, App. F.2
Support-masked and gated-local nearly tie at $k = 3$	measured	Table 5
Dense-query outputs are sparse-shaped under sparse training and track the target under dense training	measured	Fig. 7, Table 2
Errors concentrate in support selection as k grows; the decomposition does not by itself identify the class	measured	Table 14, App. H
Removing support BCE and off-support ℓ_1 jointly moves the nearest reference from gated-local to support-masked ($k{=}5$, three paired seeds, intersection-correct eligibility)	causal contrast	Table 6, App. F.2

continued on next page

Table 16 (continued)

statement	status	evidence
At k=8 the same two changes move the family coordinates in the same directions by smaller amounts: $(0.12, 0.00) \rightarrow (0.21, 0.03) \rightarrow (0.65, 0.30)$ under the paper's sampler (five seeds) and $(0.33, 0.02) \rightarrow (0.52, 0.07) \rightarrow (0.78, 0.39)$ under strict conditioning (three seeds); the objective step leaves gated-local nearest on every population; after the support step support-masked is nearest on 5/5 seeds among intersection-correct queries and on 2/5 among all queries	measured (all three eligibility populations reported; the registered switch predictions were not met)	Table 7; App. F.2
From initialization at constant fraction, the 60k reduction rises with ϵ $(0.06/0.45/0.64/0.67/0.83)$ and count-matched (fraction, steps) pairs give reductions within 0.08; on a trained network a late admixture (steps 40k–60k) reaches the uniform admixture's gap (0.015 vs 0.020), an early one (steps 2k–22k) returns to 0.108 by 60k	measured (three seeds per cell; timing arms three seeds, dense trajectories one seed)	Sec. 4, 4.3; App. G, J
An energy-matched support step under pure MSE moves the nearest reference from support-masked to global, to an interior point of the family (k=5, five paired seeds, base-correct eligibility)	causal contrast	Tables 6, 8, 11
Both training steps move the normal block's orientation beyond what its size explains: the network's change in the target contrast exceeds that of a size-matched control built from the same Jacobians, at k=5 and at k=8 under both samplers (one seed of the small objective step under the paper's k=8 sampler excepted; per-seed values in the table)	measured (exploratory; paired null)	Table 10, App. F.2
At the 60k endpoints of the timing arms the dense-query gap differs severalfold while the sparse-query projection keeps β near zero; the largest paired α shift toward support-masked equals the baseline's cross-seed range (three seeds; trajectory statement one seed); paired on identical queries, the converted and unconverted endpoints' full Jacobians differ by 0.48–0.55 floor units on three seeds, less than one undosed run's 500-step self-difference (0.59–0.63), and the dose-specific change along the onset is 0.01–0.06 (one seed)	measured (exploratory; design frozen first)	App. J
Dense-query and mixed training reach a near-global map at nearly unchanged sparse accuracy	causal contrast	App. G
A small dense admixture removes most of the sparse-to-dense gap; the response is far from proportional to the mixture weight (Sec. 4)	causal dose response	Fig. 3, App. G
Why optimization selects one extension; whether minibatch stochasticity drives its return; shared versus support-specific rows; whether the local computation on sparse queries follows the extension along a conversion trajectory (endpoints on three seeds say no; the trajectory itself is one seed)	open	Secs. 4, 7; App. F

Not claimed. The following statements are outside what the evidence supports.

- A reconstructed circuit. The diagnostic ranks the query map against three references and the two-parameter family through them, with a residual outside that family (Table 8); it does not certify a reference, and it does not say which heads or weights realize the map.
- That training minimizes the Jacobian norm. Proposition 3 explains why gated-local is a canonical reference; Sec. 7 and App. I record that the squared Frobenius-norm cost does not predict the transitions.
- An optimizer-level mechanism for the selection. The timing evidence (App. J) shows that the extension follows the current training distribution and no more.
- The masked-versus-local discrimination beyond the main setting, or any statement about overcomplete dictionaries, real data, or $k > 12$.
- That the support-selection failure is specific to the gated class: a query-linear control shows the same decomposition (App. H).
- That the three computations exhaust the behaviorally equivalent class. The global inverse is the unique query-linear exact solution (Lemma 1); the gated-local field is the unique pointwise minimum-Frobenius-norm Jacobian among the exact interpolants of Proposition 3; and the support-masked field is canonical for the shared-encoder TopK class (App. B). These characterizations pin a Jacobian field or a class representative; they do not exhaust the exact-interpolant class or identify a unique function away from M_s: adding a squared $(s{+}1)$-monomial deviation (Sec. 2.3) to any exact interpolant, the gated-local map included, preserves its values and Jacobians on M_s and changes it off M_s.

M Experimental details and reproducibility

All data are synthetic, and App. D specifies their complete generation process. The architecture, objective, metrics, evaluation-set shapes, interventions, and classical-estimator settings are given in the corresponding appendix sections. Endpoint comparisons use each run's preassigned final checkpoint. Trajectory analyses report their checkpoint schedule explicitly, and no checkpoint is selected separately to optimize an endpoint metric. Base and intervention runs do not share a universal budget or seed count, so each table or experiment description states the applicable count. The size-matched control of App. F.2 is computed from the measured Jacobians on two fresh 64×4 exact-$s{=}2$ banks (seeds 2026091301 and 2026091401), with twenty random draws per query whose random key depends on the bank seed, episode and query only, so every training arm at a given seed receives the same draws; an independent implementation recomputes every reported median from the stored Jacobians. The main architecture uses $d_{\text{model}}{=}256$, six layers, and eight heads.

N Reading guide for Figure 1

Figure 1 follows four stages. **Panel 1** shows the information available in one episode: the model observes the context matrix $\boldsymbol{X}_{ctx}$ and the query $\boldsymbol{x}_q$, while the dictionary $\boldsymbol{A}^\natural$, the context codes $\boldsymbol{S}_{ctx}$, and the query code $\boldsymbol{s}_q$ (dashed) are latent. All three candidate computations receive exactly the same context, so any difference between them is a difference in computation, not in available information. **Panel 2** shows the three candidates. The global route (red) recovers one episode inverse $\boldsymbol{P}^\natural = (\boldsymbol{A}^\natural)^{-1}$ and applies the same operator to every query. The support-masked route (violet) branches from that recovered inverse: it computes the full dense code $\boldsymbol{P}^\natural \boldsymbol{x}_q$, selects a support by TopK/thresholding, and keeps only the selected output coordinates, giving f_{mask} of Sec. 2. The support-gated local route (green) selects first and solves only the selected subproblem: it routes context evidence, infers the active support S, and applies the restricted decoder f_{local} of Sec. 2; unselected coordinates are returned as zero. The stages are a schematic of the gated-local computation, not architectural wiring claims and not a claim about how the network selects the support (Sec. 3). **Panel 3** shows the equivalence of Proposition 1: the sparse query lies on one subspace of M_s (never at the origin where the subspaces meet), all three computations act on it, and their outputs coincide (the route-colored three-term chain), so sparse-task accuracy cannot separate them. **Panel 4** previews the identification and selection logic. Top: along active tangent directions τ (drawn along the elongated M_s region) the three references respond identically ($G_S\tau = M_S\tau = \boldsymbol{P}^\natural\tau$,

the pinned block of Sec. 2.3), while a held-out normal direction v_N (drawn perpendicular to τ, with a right-angle mark at the query) produces three different outputs, drawn as response vectors: gated-local gives zero, support-masked responds on the selected rows only, and the global inverse responds on all rows (Sec. 3). The verdict box states the measured result: the sparse-trained Transformer responds like gated-local (nearest for $k \geq 5$; the support-dependent near-tie at $k = 3$ is in Table 1). "Responds like" is the diagnostic's ranking: it means the response along v_N is small, as a size-matched random response confirms (Table 9), and does not assert a select-then-decode mechanism inside the network. Bottom: the two training-design axes of Sec. 4. The header rows give the reference-level selection map (objective: gated-local to support-masked; query support under pure MSE: support-masked to global), where the two arrows summarize separately matched experiment pairs (App. F.2). The three cards give the behavioral view of the query-support interventions under the standard objective, with the gap defined as $\mathrm{gap}(f) = \langle\cos\rangle_{\mathrm{sparse}} - \langle\cos\rangle_{\mathrm{dense}}$ (zero for $\boldsymbol{P}^{\natural}$): petal width stands for the dimension of the tangent block that training pins, the blue dots mark rare full-dimensional anchors, and the quoted gap values are the measured reductions (the $s'{:}2{\to}3$ step removes about 96% at $k{=}5$, and the 0.5% dense admixture removes about 82%). The figure describes functional behavior. It does not claim that the network stores $\boldsymbol{P}^{\natural}$ or an explicit TopK or pseudoinverse routine in any particular module.