WHAT MAKES ONE-STEP INFERENCE WORK? A DECOMPOSITION OF GRADIENT ORIGIN NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Gradient Origin Networks remove the encoder, instead inferring a latent code with a single gradient step through the decoder with respect to the origin. We ask what a decoder must learn for such a restricted inference rule to work. We find that autoencoder decoders often reconstruct competitively when their latent codes are optimised directly, yet perform poorly under the same one-step inference rule. The difference therefore lies in inference rather than reconstruction capacity. In the linear case this has an exact explanation: one-step inference uses the decoder transpose while optimal inference uses its pseudoinverse, with the two coinciding when the decoder Gram matrix is isotropic up to scale. An untied autoencoder need not learn this parametrisation because its encoder can absorb arbitrary invertible changes of decoder coordinates. Nonlinear GONs similarly learn flatter origin-Jacobian spectra, but imposing this geometry on an autoencoder is not sufficient. Instead, successful one-step inference depends on training reconstruction through the decoder's own inference map. A decoder can reconstruct well and still be poorly suited to gradient inference.

1 INTRODUCTION

Latent-variable models must solve an inference problem: given an observation x, find a latent code z from which a decoder F can reconstruct it. Autoencoders and variational autoencoders solve this problem through amortisation: a trained neural network predicts the code or posterior parameters in a single forward pass (Kingma & Welling, 2014; Rezende et al., 2014). Optimisation-based models instead infer the code by directly minimising reconstruction error, as in sparse coding, predictive coding, and latent-optimisation generators (Olshausen & Field, 1996; Rao & Ballard, 1999; Bojanowski et al., 2018). Semi-amortised and iterative-amortised methods interpolate between these extremes by refining an encoder prediction with gradient steps (Kim et al., 2018; Marino et al., 2018). The question of what the generator learns in response to its inference scheme is therefore not incidental: amortisation gaps, truncation errors, and inner-loop approximations can all become properties of the trained generative model (Cremer et al., 2018).

Gradient Origin Networks (GONs) (Bond-Taylor & Willcocks, 2021) are optimisation-based models. They remove the encoder, initialise the latent at a fixed origin $z_0 = 0$, take one gradient step on reconstruction error, and decode from the resulting code. For squared reconstruction loss, the inferred code is

$$z_1 = J_0^\top (x - F(0)), \qquad J_0 = \left.\frac{\partial F}{\partial z}\right|_{z=0}.$$

Thus all information used for inference passes through a single local linear map: the decoder Jacobian at the origin. There is no independently learned recognition network and no iterative latent search. The central question of this paper is therefore: *what must the decoder learn so that this inference rule works?*

A circular answer is immediate: GONs are trained using the one-step rule, so they learn decoders for which it is useful. But this does not explain what property has been learned, why an ordinary autoencoder decoder need not have it, or whether the property can be imposed directly. We ask the following three questions: First, does the decoder have enough capacity to reconstruct the input from some latent code? Second, does the origin-gradient rule find such a code? Third, what features of the decoder make the one-step rule more effective?

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.

Our starting point is a reconstruction-space decomposition. For a fixed decoder, we compare the one-step code z_1 with an optimised latent reference

$$z^\star = \arg\min_z \mathrm{MSE}(F(z), x),$$

and measure the reconstruction-space inference gap

$$\Delta_{\mathrm{MSE}} = \mathrm{MSE}(F(z_1), x) - \mathrm{MSE}(F(z^\star), x).$$

This asks whether the decoder can reconstruct well from *some* code, and separately whether the origin-gradient rule finds such a code. Empirically, autoencoder decoders often reconstruct competitively under optimised latent inference, but their origin-gradient codes retain large reconstruction-space gaps. The difference is therefore not in the decoder's capacity, but in inference from the gradient.

The linear case makes the mechanism exact. For $F(z) = Wz$, one-step inference uses the weight-transpose $W^\top x$, while optimised inference uses the least-squares code involving the pseudoinverse of W. These coincide for all inputs when the decoder Gram matrix is isotropic, $W^\top W \propto I$. This is the same condition under which the decoder transpose acts as its pseudoinverse, familiar from tight frames and sparse-recovery analyses (Casazza & Kutyniok, 2013; Donoho et al., 2006). A standard untied linear autoencoder has no reason to select this parametrisation: its reconstruction objective depends on the decoder subspace but is invariant to invertible changes of basis within that subspace, a symmetry closely related to the classical linear-autoencoder/PCA analysis (Baldi & Hornik, 1989; Kunin et al., 2019; Oftadeh et al., 2020). The autoencoder can learn an encoder that absorbs the inverse basis transform; the GON cannot, because its implicit encoder is tied to the decoder transpose.

In nonlinear decoders, the analogous object is the origin-Jacobian Gram matrix $J_0^\top J_0$. We show that trained GONs have flatter, better-conditioned origin-Jacobian spectra than autoencoders, and then use interventions to test whether this geometry causes good inference, or whether it is a by-product. Conditioning and code distribution are insufficient on their own, and raw manifold-alignment diagnostics do not by themselves explain the gap. In nonlinear models, the gradient pathway through the inferred code is necessary. The resulting explanation is therefore limited: successful one-step inference requires reconstruction training to be coupled to the decoder's own origin-Jacobian inference map.

Our contributions are:

1. **A capacity–inference decomposition.** We introduce an optimised-code reference that separates decoder capacity from the quality of one-step origin-gradient inference, showing that autoencoder decoders can have competitive reconstruction capacity while still failing under the GON inference rule.
2. **An exact linear account.** We prove that one-step linear inference requires that the decoder Gram matrix is isotropic, and connect this to the known gauge freedom of untied autoencoders: the AE objective is invariant to decoder reparametrisations that change the conditioning required by one-step inference.
3. **A nonlinear geometric characterisation and intervention study.** We analyse origin-Jacobian spectra and use controlled interventions to distinguish characteristics that cause from those that correlate, localising the nonlinear mechanism while leaving a complete closed-form account of nonlinear training dynamics open.

The paper is not a claim that GONs are universally preferable to amortised inference, nor a complete theory of nonlinear decoder training. Its aim is narrower: to identify what one-step encoder-free inference produces, why ordinary autoencoders need not learn it, and which parts of the GON training setup matter most.

2 Setup and Metric

We study decoders $F_\theta : \mathbb{R}^{n_z} \to \mathbb{R}^D$ that reconstruct an observation $x \in \mathbb{R}^D$ from a latent code z. A standard autoencoder additionally learns an encoder E_ϕ and reconstructs as $F_\theta(E_\phi(x))$. A GON removes this encoder. Starting from $z_0 = 0$, one gradient step on

$$\mathcal{L}(z; \theta, x) = \tfrac{1}{2}\|F_\theta(z) - x\|^2 \tag{1}$$

gives

$$z_1 = \alpha J_0^\top \big(x - F_\theta(0)\big), \qquad J_0 = \left.\frac{\partial F_\theta}{\partial z}\right|_{z=0}. \tag{2}$$

Unless varied explicitly, we absorb the scalar step size into the inference scale and write

$$z_1 = J_0^\top \big(x - F_\theta(0)\big), \tag{3}$$

with local latent Gram matrix

$$M = J_0^\top J_0. \tag{4}$$

In the linear case $F_W(z) = Wz$, $J_0 = W$ and $M = W^\top W$.

To separate decoder capacity from one-step inference quality, we evaluate every trained decoder using both the one-step code z_1 and an optimised latent reference

$$z^\star = \arg\min_z \mathrm{MSE}\big(F_\theta(z), x\big), \tag{5}$$

computed with the decoder fixed and latent optimisation initialised from the same origin. We report

$$\mathrm{MSE}_{z^\star} = \mathrm{MSE}\big(F_\theta(z^\star), x\big), \qquad \mathrm{MSE}_{z_1} = \mathrm{MSE}\big(F_\theta(z_1), x\big), \tag{6}$$

and use the reconstruction-space inference gap

$$\Delta_{\mathrm{MSE}} = \mathrm{MSE}_{z_1} - \mathrm{MSE}_{z^\star}. \tag{7}$$

A small gap means the origin-gradient rule reaches a reconstruction close to the decoder's optimised latent floor; a large gap means the decoder may have useful capacity that the one-step rule fails to access. We apply the same rule in Eq. 3 to GON and autoencoder decoders, including AE decoders evaluated outside their own encoder path.

$\mathrm{MSE}_{z^\star}$, and therefore the reference-code optimiser, is found empirically through optimisation and is therefore not perfect. To minimise this, we use specific experimental rules. The latent optimisation is run with the decoder fixed, from the same origin as the one-step rule, and is monitored with both loss-plateau and gradient criteria. Capped or plateaued cells are flagged and audited with larger budgets and a long Adam baseline. However, where there is residual under-convergence, the bias is conservative: it overestimates $\mathrm{MSE}_{z^\star}$ and therefore *reduces* the measured AE inference gap. Full diagnostics are in Appendix B.1. Optimiser, convergence, scale, seed, and architecture details are deferred to Appendix B.

As a brief note on figure discipline, we use blue graphics to denote GON results and red graphics to denote AE results. Error bars or shading are used to denote ± 1 SD.

3 Distinguishing Inference From Capacity

Figure 1 decomposes reconstruction error for GON and AE decoders across datasets and latent dimensions. The optimised-code error $\mathrm{MSE}_{z^\star}$ is broadly comparable between the families. In several cells the AE floor is equal or lower, but the one-step gap separates them: GON decoders have near-zero Δ_{MSE}, whereas AE decoders retain a substantial gap.

The AE failure is therefore not that its decoder cannot represent good reconstructions; under stronger latent inference it often does. The failure is that $J_0^\top (x - F(0))$ is not a good inference map for that decoder. Equivalently, the AE encoder can learn codes in regions where the decoder reconstructs well without pressuring the decoder's origin Jacobian to produce those codes.

A natural objection is that the decomposition is preordained: a GON is trained to reconstruct from z_1, and $z^\star$ is by construction the best the decoder can do from any code, so a well-trained GON *must* have $\Delta_{\mathrm{MSE}} \approx 0$. We grant this direction of the comparison; that the GON closes its own gap amounts to a restatement of its training objective. Figure 1 shows something different, though. First, that the *autoencoder* retains a large Δ_{MSE} is not implied by its training: the AE objective never references the origin-gradient code $z_1 = J_0^\top (x - F(0))$, and nothing about minimising reconstruction through a learned encoder predicts that this map should be a *poor* inference rule for the resulting decoder. Second, the gap is only interpretable because $\mathrm{MSE}_{z^\star}$ independently certifies that the AE decoder *can* reconstruct. Its optimised-code floor is comparable to the GON's, and in several cells

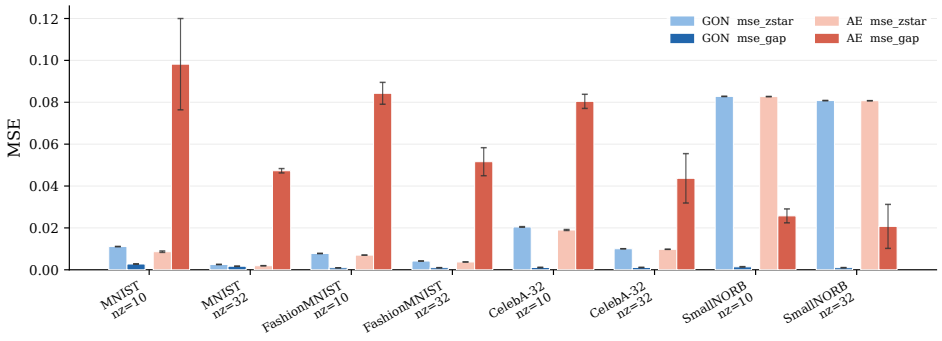
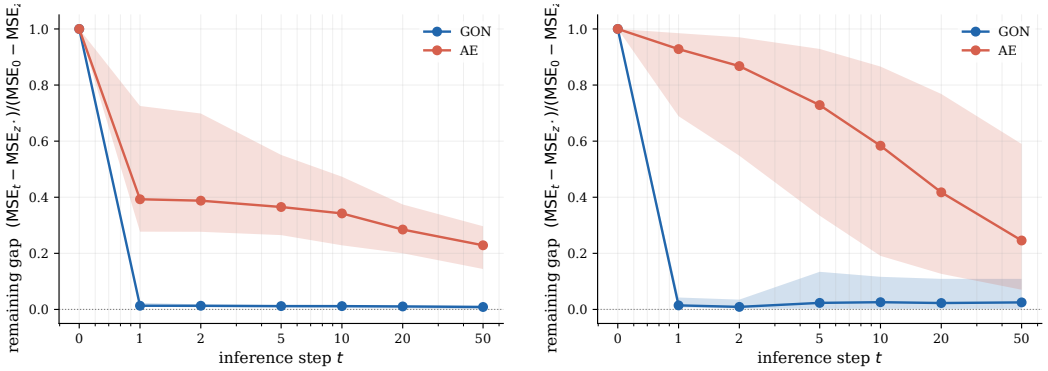


Figure 1: Decomposition of reconstruction error into decoder capacity and inference quality. Light bars show $\mathrm{MSE}_{z^\star}$, the reconstruction error obtained after optimising the latent code with the decoder fixed. Dark bars show Δ_{MSE}, the additional error incurred by using the one-step code z_1 rather than the optimised code $z^\star$. GON and AE decoders have comparable optimised-code reconstruction error, but differ sharply in one-step inference gap.



(a) MLP

(b) Convolutional

Figure 2: Fraction of the origin gap remaining along the inference trajectory. From $z_0 = 0$ we iterate the same origin-gradient rule and plot $(\mathrm{MSE}_{z_t} - \mathrm{MSE}_{z^\star})/(\mathrm{MSE}_{z_0} - \mathrm{MSE}_{z^\star})$ against step t. One step removes about 99% of the GON's gap for both architectures; for the autoencoder it removes much less and a substantial gap remains after additional steps.

lower (Figure 1), so the deficit can be found specifically in inference rather than capacity. Our results show that autoencoders are actively poor at finding a performant code with this inference mechanism and the interventions of Section 6 show that this deficit resists every correction we apply.

Iterating the origin-gradient rule rather than stopping after one step makes the decomposition visible dynamically (Figure 2). GONs close almost all of the origin-to-$z^\star$ gap immediately, whereas AEs improve much more slowly and remain far from the optimised floor after additional steps. This directly rebuts the simplest "just take a few more steps" explanation: the AE decoder may contain a good reconstruction floor, but the gradient field induced by its origin Jacobian does not efficiently access it.

These results motivate our study of the two models. The next section provides an exact account in linear models, while the subsequent sections cover non-linear models.

4 Linear Case: Exact Theory and Validation

We apply standard least-squares and linear autoencoder results to the GON inference rule, then decompose its reconstruction loss into projection and inference error. Let the decoder be linear, $F_W(z) = Wz$, with $W \in \mathbb{R}^{D \times n_z}$ full column rank and centred data. For squared loss and unit

inference scale,

$$z_1 = W^\top x, \qquad \hat{x}_1 = W z_1 = W W^\top x. \tag{8}$$

For the same decoder, optimised latent inference gives the least-squares code

$$z^\star = (W^\top W)^{-1} W^\top x, \qquad \hat{x}^\star = P_W x, \qquad P_W = W(W^\top W)^{-1} W^\top. \tag{9}$$

Thus one-step inference replaces the orthogonal projector P_W by $WW^\top$, and

$$z^\star - z_1 = \big[(W^\top W)^{-1} - I\big] W^\top x. \tag{10}$$

The discrepancy is governed entirely by the decoder Gram matrix.

Proposition 1 (Exactness of one-step linear inference)**.** *Assume W has full column rank. One-step inference is exact for all $x \in \mathbb{R}^D$, in the sense that $z_1 = z^\star$ and $WW^\top x = P_W x$, if and only if $W^\top W = I$. With free scalar step size α, the scale-invariant condition is $W^\top W \propto I$.*

Proposition 2 (Autoencoder invariance to decoder conditioning)**.** *For a linear untied autoencoder with encoder E and decoder W, the optimised reconstruction loss over E depends only on* $\mathrm{col}(W)$*. In particular, for any invertible $A \in \mathbb{R}^{n_z \times n_z}$,*

$$\min_E \mathcal{L}_{\mathrm{AE}}(E, WA) = \min_E \mathcal{L}_{\mathrm{AE}}(E, W), \tag{11}$$

even though $(WA)^\top (WA) = A^\top (W^\top W) A$ can have arbitrarily different conditioning from $W^\top W$.

The contrast is the core linear mechanism. An untied AE can learn an encoder that absorbs any invertible reparametrisation of the decoder basis; the reconstruction objective constrains the subspace but not the coordinate system within it. A GON has no separate encoder: its implicit encoder is the decoder transpose. Substituting $z_1 = W^\top x$ into the reconstruction loss gives

$$\mathcal{L}_{\mathrm{GON}}(W) = \tfrac{1}{2}\mathbb{E}_x \|x - WW^\top x\|^2. \tag{12}$$

Writing $W = UA$ with $U^\top U = I$ gives

$$\mathcal{L}_{\mathrm{GON}}(UA) = \underbrace{\tfrac{1}{2}\mathbb{E}_x \|x - UU^\top x\|^2}_{\text{projection error}} + \underbrace{\tfrac{1}{2}\mathbb{E}_x \|(I - AA^\top)U^\top x\|^2}_{\text{inference error}}. \tag{13}$$

For a fixed subspace, the second term vanishes exactly when the decoder basis is orthonormal up to the inference scale. Full proofs are given in Appendix G.

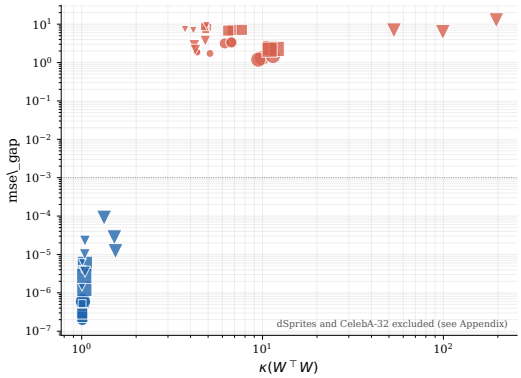


Figure 3: Empirical validation in the linear setting. Each point is a trained linear model; the horizontal axis is the decoder Gram condition number $\kappa(W^\top W)$ and the vertical axis is the one-step inference gap Δ_{MSE} (both log scale). Linear GONs cluster near $\kappa(W^\top W) \approx 1$ with near-zero gap, while linear AEs recover comparable reconstruction floors with worse conditioning and large gaps.

Figure 3 validates the theory. Both families recover useful reconstruction subspaces. For an untied AE, however, all decoder bases WA related by an invertible latent transformation are equivalent under reconstruction loss, since the encoder can compensate by the inverse transformation. The

decoder Gram matrix therefore varies freely across reconstruction-equivalent parametrisations. In a GON this invariance is broken: because inference is fixed to $z_1 = W^\top x$, changing the decoder basis also changes the inferred code. The objective consequently selects the near-isometric parametrisation required for one-step inference to agree with least squares.

A linear detach ablation sharpens this interpretation (Appendix G). If $z_1 = W^\top x$ is computed in the forward pass but detached before backpropagation, the model still learns $\kappa(W^\top W) \approx 1$ and near-zero gaps. Thus, in the linear case, forward tying through $W^\top x$ is sufficient: the objective itself penalises non-isometric parametrisations. This will differ in the nonlinear case, where $z_1 = J_0^\top (x - F(0))$ depends on the parameters through a Jacobian whose training pathway is needed.

5 Nonlinear Decoder Geometry

In nonlinear decoders, we rather analyse the origin-Jacobian Gram matrix $M = J_0^\top J_0$. This follows from the first-order approximation $F(z) \approx F(0) + J_0 z$: around the origin, J_0 plays the role of the linear decoder matrix, and the one-step update rule therefore has the same form as $z_1 = W^\top x$. Because the one-step code in Eq. 3 passes entirely through $J_0^\top$, successful one-step inference should require a high-rank, well-conditioned origin Jacobian. Let $\sigma_i(J_0)$ denote the singular values of J_0 and define the scale-invariant participation ratio

$$\mathrm{pr}(J_0) = \frac{(\sum_i \sigma_i)^2}{\sum_i \sigma_i^2}. \tag{14}$$

We use this only as a summary of spectral flatness; thresholding choices and alternative rank summaries are reported in Appendix I.

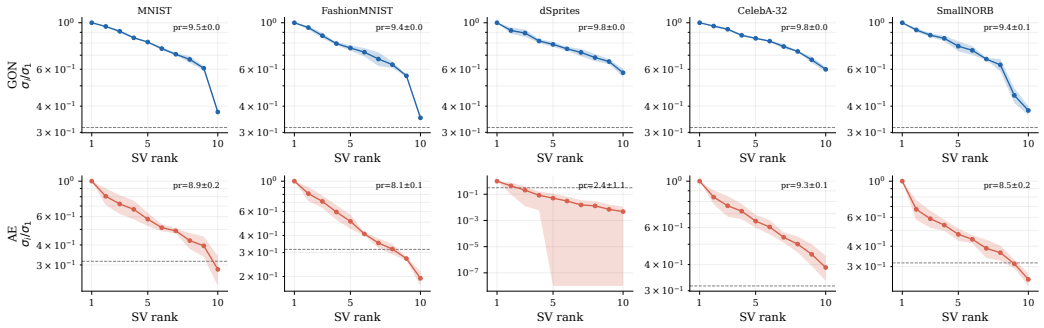


Figure 4: Singular-value spectra of the origin Jacobian J_0 for trained nonlinear GON (top, blue) and AE (bottom, red) decoders across five datasets, normalised by the leading singular value, with participation ratio annotated (mean ± 1 SD over seeds). GON spectra are near-flat with high participation ratio; AE spectra decay more steeply and have weaker minimum-gain directions, even when their participation ratio remains moderately high.

Figure 4 shows the nonlinear analogue of the linear condition. GON decoders have comparatively flat J_0 spectra, so the local map used for inference preserves usable scale across latent directions. AE decoders instead have worse tail conditioning: the origin-gradient update receives strong signal in a few dominant directions and weak, noise-sensitive signal in the rest. This does not mean the AE decoder has low capacity; Section 3 has already shown that optimised latent inference can still reconstruct competitively. The failure recorded is that the coordinate system around the origin is not organised specifically for $J_0^\top (x - F(0))$.

A related diagnostic is the alignment between $\mathrm{col}(J_0)$ and a PCA approximation to the data manifold. We defer the full principal-angle plots to Appendix I.2 since the result is mixed: convolutional GONs often show stronger alignment, while MLP differences are smaller and sometimes comparable. Alignment therefore does not by itself explain the separation. Together with the spectral results above, this suggests that several aspects of the origin Jacobian differ between the trained models, but these observations are merely descriptive. However, these differences motivate targeted interventions on the observed geometry that are carried out in the next section.

Intervention / reference	Δ_{MSE}	$\mathrm{MSE}_{z^\star}$	$\kappa(M)$	pr	align
GON (reference)	0.003	0.011	7.1	6.1	0.34
Autoencoder (reference)	0.098	0.008	14.1	3.9	0.33
AE + J_0-isometry reg. (strong λ)	0.149	0.053	1.1	9.7	–
Fresh decoder on frozen GON codes	0.079	0.010	17.3	–	–
Detach z_1 (vs. full GON 0.003)	0.009	0.019	7.2	–	–
Freeze GON decoder, refit encoder	0.002	0.010	–	6.2	0.39
then unfreeze under AE loss	0.060	0.008	–	4.5	0.32

Table 1: Intervention summary on MNIST, $n_z = 10$, mean over three seeds. No static property reproduces the GON gap. Regularising an AE's J_0 toward isometry improves $\kappa(M)$ and pr but worsens both Δ_{MSE} and $\mathrm{MSE}_{z^\star}$; a fresh decoder fit to frozen GON codes reconstructs them but has an AE-like origin-Jacobian gap; detaching z_1 worsens the nonlinear GON; and a frozen GON decoder supports the original codes, but the structure drifts under AE training once the decoder is unfrozen.

6 Interventions: What Is Necessary and What Is Not Sufficient

A well-conditioned origin Jacobian is the signature of the trained GON, but several simpler explanations remain possible: perhaps conditioning alone is enough; perhaps the GON code distribution is enough; perhaps the geometry, once acquired, is stable under ordinary AE training; or perhaps the nonlinear gradient pathway through J_0 is incidental. We test these with interventions that isolate one explanation at a time. Table 1 gives the compact summary; Figures 5 and 6 show the two most important dynamics.

Conditioning alone is not sufficient. We train AEs with a regulariser on the origin Gram matrix $M = J_0^\top J_0$, penalising deviation from an isometric form, the direct nonlinear analogue of imposing $W^\top W \approx I$. The regulariser improves the conditioning of M, in places matching or exceeding that of GONs, but it does not reliably reduce Δ_{MSE} to the GON regime: on MNIST and FashionMNIST stronger regularisation worsens the gap, and on CelebA-32 and SmallNORB it only partly closes it. The diagnostic is $\mathrm{MSE}_{z^\star}$, which degrades when the penalty is strong: the decoder becomes worse even under optimised inference. The regulariser can make J_0 look GON-like without making $J_0^\top(x - F(0))$ the right inference map. This confirms matching the conditioning of $J_0^\top J_0$ is insufficient to reproduce GON-level one-step inference.

Code distribution alone is not sufficient. We next ask whether it is the codes z_1 that matter, or whether a GON must produce them by its own inference mechanism. We train a fresh decoder to reconstruct from frozen GON codes, supplied externally rather than generated by its own $J_0^\top(x - F(0))$. The decoder fits the codes, its optimised-code error matches the GON's, but its own origin Jacobian does not become a usable one-step map: $\kappa(M)$ remains AE-like and Δ_{MSE} stays far above the GON. Exposure to good codes is not enough. A decoder fit to supplied codes can learn a flexible function, however a decoder with the GON's one-step inference rule is constrained.

Self-consistency is not self-sustaining. Is the geometry, once acquired, retained under ordinary AE training? We freeze a trained GON decoder and refit a fresh encoder through it: the encoder recovers codes aligned with the original GON one-step codes and reaches near-zero Δ_{MSE}. The decoder thus contains genuine self-consistent structure; this is not an artefact of evaluating the model with its own inference rule. When we instead unfreeze the decoder under standard reconstruction training, reconstruction stays good but the geometry is not preserved: the origin-Jacobian structure drifts back toward the AE regime and the link to the GON codes decays. This is exactly the prediction of the autoencoder invariance: reconstruction loss alone does not identify the parametrisation one-step inference needs.

The Jacobian-gradient pathway is necessary in the nonlinear case. Because J_0 depends on decoder parameters, the nonlinear loss has a gradient pathway through the inferred code z_1 as well

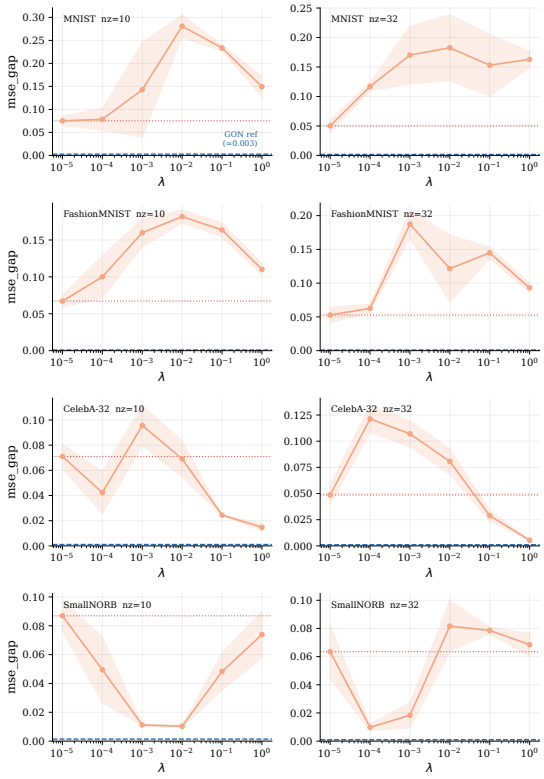


Figure 5: Explicit J_0-isometry regularisation of AE decoders. The regulariser improves origin-Jacobian conditioning but does not recover GON-level one-step inference; in several cells, stronger regularisation worsens the reconstruction-space gap or damages the optimised-code floor.

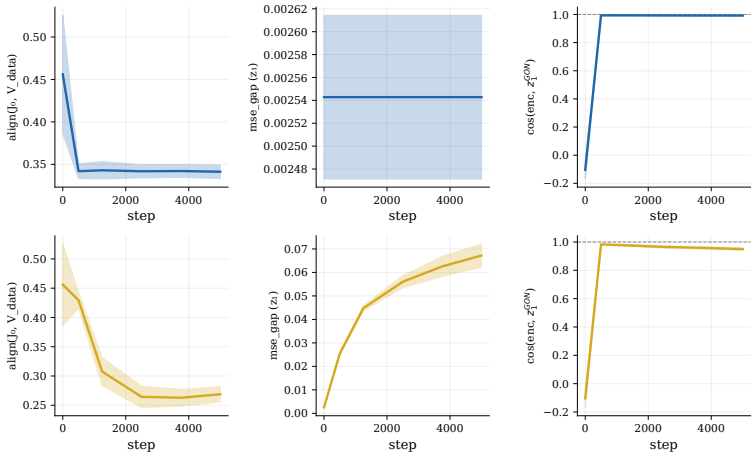


Figure 6: GON-to-AE transfer. With the GON decoder frozen, a fresh encoder recovers the original GON one-step codes and preserves near-zero Δ_{MSE}. With the decoder unfrozen under standard reconstruction training, reconstruction stays good but the origin-Jacobian geometry drifts toward the AE regime and the alignment with the GON codes decays.

as through the decoder evaluation $F(z_1)$. We detach z_1 during backpropagation, keeping it in the forward pass but removing the gradient through its dependence on the parameters. The ablation consistently worsens Δ_{MSE} and changes the learned geometry. This does not prove that the second-order term is the unique cause of the self-consistent state; it establishes the more precise claim that

the gradient pathway through J_0 is required for the nonlinear geometry and for one-step inference. This contrasts with the linear detach result, where forward tying alone suffices. The effect generalises across datasets: detaching z_1 raises Δ_{MSE} relative to the full-GON baseline by factors of roughly 1.5 to 11 on MNIST, FashionMNIST, CelebA-32, and SmallNORB (over $n_z \in \{10, 32\}$ and three seeds), so the gradient pathway is important on each; only on dSprites at $n_z = 10$, the synthetic outlier whose GON gap is itself large, does the effect fall within seed noise.

What the interventions establish. No single intervened-upon property tested here reproduces GON one-step inference. A well-conditioned origin Jacobian, a favourable externally supplied code distribution, or good reconstruction loss can each be present without recovering the GON regime. The effect instead requires a self-consistent coupling: reconstruction quality must be tied to the codes produced by the decoder's own origin Jacobian, and in nonlinear models the gradient pathway through that Jacobian is needed to maintain this coupling during training. These conclusions are not specific to the representative MNIST cell: Table 5 repeats the interventions across four of the five datasets, with the same qualitative outcome. Neither conditioning nor code distribution reaches the GON regime on any of them, and the gradient-pathway ablation generalises as noted above. dSprites serves as an exception to the general rule as model capacity becomes an issue, making the results uninformative. Cross-dataset results for the static-property interventions and further details of the detach ablation are given in Appendix J.

7 Limitations and Conclusion

This paper is a decomposition and localisation of one-step inference rather than a complete theory of nonlinear GON training. We show that the empirical GON–AE separation is primarily an inference effect, prove the linear mechanism, identify the AE invariance that leaves decoder coordinates underconstrained, characterise the nonlinear trained state through J_0, and rule out several natural sufficient explanations.

The remaining open problem is the nonlinear weight-space mechanism. Writing

$$z_1(x) = J_0^\top (x - F(0)),$$

a trained GON approximately satisfies a coupled condition: the origin Jacobian is both the local linearisation of F at the origin and an approximate inverse map for the data under the decoder. In the linear case these roles collapse to the single condition $W^\top W \approx I$ (up to the inference scale). In the nonlinear case, however, J_0 determines the code z_1, while reconstruction is evaluated at $F(z_1)$. Good one-step inference therefore requires the decoder's local map at the origin to produce codes that land in regions where the decoder reconstructs well. A flat J_0 spectrum or an explicit isometry regulariser does not itself guarantee this compatibility.

Our detach result indicates that this coupling is enforced through a curvature-sensitive training pathway: gradients flow not only through the reconstruction map $F(z_1)$ but through the construction of z_1 itself, and hence through the dependence of J_0 on the decoder parameters. That observation narrows down where the mechanism may lie, but it does not yet provide a closed-form dynamical account of how gradient descent reaches the self-consistent fixed point. We therefore do not claim a complete nonlinear theory analogous to the linear propositions.

The practical implication is that cheap gradient inference is not guaranteed by decoder capacity alone. A learned encoder can hide arbitrary decoder reparametrisations that later make one-step or few-step latent optimisation ineffective. GON training works because it removes that freedom: reconstruction quality is trained through the decoder's own inference rule. That is the mechanism-level constraint this paper identifies.

AI use statement

We used generative AI tools (a large language model assistant) for the following tasks in the *required-disclosure* categories: help develop theoretical models or conceptual frameworks; formulate the mathematical claims; provide critical ingredients for proving them; assist in the writing of the proofs' propose or refine hypotheses; design the research methodology or experiments; implement the methods.

We did *not* use generative AI tools for: generating synthetic datasets, translation, cleaning and reformatting datasets, qualitative or thematic data analysis, interpreting results.

In the *recommended-disclosure* categories, we used generative AI tools to edit and reorganise the released research code and to modify scientific figures.

We have reviewed all AI-assisted work. We used AI tools to assist with algebraic derivations and proof development for the linear-algebra results in Section 4, with most being standard in the literature anyway but the GON-specific derivations and Equation 10 were derived by the authors.

AI played a larger role in the production of code, although the codebase and the experiments were reviewed by the authors.

We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Guillaume Alain and Yoshua Bengio. What regularized auto-encoders learn from the data-generating distribution. *Journal of Machine Learning Research*, 15:3563–3593, 2014.

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. Deep equilibrium models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Pierre Baldi and Kurt Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2(1):53–58, 1989.

Nitin Bansal, Xiaohan Chen, and Zhangyang Wang. Can we gain more from orthogonality regularizations in training deep networks? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

Piotr Bojanowski, Armand Joulin, David Lopez-Paz, and Arthur Szlam. Optimizing the latent space of generative networks. In *International Conference on Machine Learning (ICML)*, 2018.

Sam Bond-Taylor and Chris G. Willcocks. Gradient origin networks. In *International Conference on Learning Representations (ICLR)*, 2021. arXiv:2007.02798.

Hervé Bourlard and Yves Kamp. Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics*, 59(4–5):291–294, 1988.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations (ICLR)*, 2019.

Peter G. Casazza and Gitta Kutyniok (eds.). *Finite Frames: Theory and Applications*. Birkhäuser, 2013.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15750–15758, 2021.

Moustapha Cisse, Piotr Bojanowski, Edouard Grave, Yann Dauphin, and Nicolas Usunier. Parseval networks: Improving robustness to adversarial examples. In *International Conference on Machine Learning (ICML)*, pp. 854–863, 2017.

Chris Cremer, Xuechen Li, and David Duvenaud. Inference suboptimality in variational autoencoders. In *International Conference on Machine Learning (ICML)*, pp. 1078–1086, 2018.

Ingrid Daubechies, Michel Defrise, and Christine De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57(11):1413–1457, 2004.

David L. Donoho, Michael Elad, and Vladimir N. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on Information Theory*, 52(1): 6–18, 2006.

Simon S. Du, Wei Hu, and Jason D. Lee. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *International Conference on Machine Learning (ICML)*, pp. 399–406, 2010.

Suriya Gunasekar, Blake Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nathan Srebro. Implicit regularization in matrix factorization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Like Hui, Mikhail Belkin, and Preetum Nakkiran. Limitations of neural collapse for understanding generalization in deep learning. *arXiv preprint arXiv:2202.08384*, 2022.

Ilyes Khemakhem, Diederik P. Kingma, Ricardo Pio Monti, and Aapo Hyvärinen. Variational autoencoders and nonlinear ICA: A unifying framework. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020.

Yoon Kim, Sam Wiseman, Andrew C. Miller, David Sontag, and Alexander M. Rush. Semi-amortized variational autoencoders. In *International Conference on Machine Learning (ICML)*, 2018.

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning (ICML)*, pp. 3519–3529, 2019.

Daniel Kunin, Jonathan M. Bloom, Aleksandrina Goeva, and Cotton Seed. Loss landscapes of regularized linear autoencoders. In *International Conference on Machine Learning (ICML)*, pp. 3560–3569, 2019.

Daniel Kunin, Javier Sagastuy-Brena, Surya Ganguli, Daniel L. K. Yamins, and Hidenori Tanaka. Neural mechanics: Symmetry and broken conservation laws in deep learning dynamics. In *International Conference on Learning Representations (ICLR)*, 2021.

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 *International Conference on Machine Learning (ICML)*, pp. 4114–4124, 2019.

Joseph Marino. Predictive coding, variational autoencoders, and biological connections. *Neural Computation*, 34(1):1–44, 2022.

Joseph Marino, Yisong Yue, and Stephan Mandt. Iterative amortized inference. In *International Conference on Machine Learning (ICML)*, pp. 3403–3412, 2018.

Beren Millidge, Alexander Tschantz, and Christopher L. Buckley. Predictive coding approximates backprop along arbitrary computation graphs. *Neural Computation*, 34(6):1329–1368, 2022. arXiv:2006.04182.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2018.

Thomas Moreau and Joan Bruna. Understanding trainable sparse coding via matrix factorization. In *International Conference on Learning Representations (ICLR)*, 2017.

Reza Oftadeh, Jiayi Shen, Zhangyang Wang, and Dylan Shell. Eliminating the invariance on the loss landscape of linear autoencoders. In *International Conference on Machine Learning (ICML)*, pp. 7405–7413, 2020.

Erkki Oja. Simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, 15(3):267–273, 1982.

Bruno A. Olshausen and David J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996.

Vardan Papyan, X. Y. Han, and David L. Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40): 24652–24663, 2020.

Jeffrey Pennington, Samuel S. Schoenholz, and Surya Ganguli. Resurrecting the sigmoid in deep learning through dynamical isometry: Theory and practice. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Elad Plaut. From principal subspaces to principal components with linear autoencoders. *arXiv preprint arXiv:1804.10253*, 2018.

Rajesh P. N. Rao and Dana H. Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2(1):79–87, 1999.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning (ICML)*, pp. 1278–1286, 2014.

Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In *International Conference on Machine Learning (ICML)*, 2011.

Amartya Sanyal, Philip H. S. Torr, and Puneet K. Dokania. Stable rank normalization for improved generalization in neural networks and GANs. In *International Conference on Learning Representations (ICLR)*, 2020.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19(70): 1–57, 2018.

Michael Tschannen, Josip Djolonga, Paul K. Rubenstein, Sylvain Gelly, and Mario Lucic. On mutual information maximization for representation learning. In *International Conference on Learning Representations (ICLR)*, 2020.

James C. R. Whittington and Rafal Bogacz. An approximation of the error backpropagation algorithm in a predictive coding network with local Hebbian synaptic plasticity. *Neural Computation*, 29(5):1229–1262, 2017.

A Code Reproducibility

Attached in the supplementary material is the code. You will find all details for reproduction in README.md and REPRODUCE.md, where there are two options. If you wish to regenerate figures/tables from the results, you can run path A. If you wish to retrain the full pipeline, you can run path B.

B Experimental Protocol and Evaluation Details

Datasets and preprocessing. We use five image datasets: MNIST, FashionMNIST, dSprites, CelebA-32, and SmallNORB. Every image is resized to 32×32 and converted to a tensor with pixel values in $[0, 1]$; no further normalisation is applied, and decoders terminate in a sigmoid so their outputs lie in the same range. MNIST, FashionMNIST, and SmallNORB are single-channel; CelebA-32 is three-channel; dSprites is loaded from its packaged array and bilinearly resized. We use up to $60{,}000$ training images and the standard test split (up to $10{,}000$ images) per dataset.

Decoder and encoder architectures. All models share a decoder backbone per architecture; the GON adds no encoder, while the autoencoder adds a matching encoder.

- **MLP decoder:** $\mathrm{Linear}(n_z \to 128)$, then a stack of $\mathrm{GroupNorm}$–SiLU–$\mathrm{Linear}(128 \to 128)$ blocks, then $\mathrm{Linear}(128 \to D)$ and a sigmoid, where D is the flattened image dimension. The MLP autoencoder encoder mirrors this ($\mathrm{Linear}(D \to 128)$, GroupNorm, SiLU, $\mathrm{Linear}(128 \to n_z)$).
- **Convolutional decoder:** a linear projection $\mathrm{Linear}(n_z \to 32 \cdot s \cdot s)$ reshaped to 32 channels at spatial size s, followed by GroupNorm–SiLU and two 3×3 convolutions (padding 1), ending in a sigmoid. The convolutional autoencoder encoder is the transpose-shaped path (3×3 convolutions, GroupNorm, SiLU, a linear map to n_z).

Normalisation is $\mathrm{GroupNorm}$ (up to 8 groups) and the nonlinearity is SiLU throughout. Hidden width is 128 for the MLP and 32 channels for the convolutional model. We sweep latent dimension $n_z \in \{10, 32, 128\}$.

Training. Both families minimise the same squared reconstruction loss, $\frac{1}{2}\|F_\theta(z) - x\|^2$ (per-sample sum over pixels, averaged over the batch), differing only in how z is produced: the autoencoder uses its encoder, the GON uses the one-step origin-gradient code z_1 equation 3 with inference scale $\alpha = 1$ and a single step in the loop. We train with Adam (learning rate 10^{-3}), batch size 128, for $20{,}000$ steps, with three random seeds $\{3, 4, 5\}$ per configuration (Section B.2); the seed sets parameter initialisation and minibatch order. Reconstruction loss on the test split is logged during training and used only for monitoring—there is no separate validation split, and no early stopping or hyperparameter selection is performed on it. This yields 5 datasets $\times$ 2 architectures $\times$ 3 n_z $\times$ 3 seeds $\times$ 2 families $= 180$ trained decoders.

Latent inference and the origin Jacobian. At evaluation, every decoder is probed with the same origin-gradient rule equation 3: $z_1 = \alpha J_0^\top (x - F(0))$ at $\alpha = 1$, where the origin Jacobian $J_0 = \partial F / \partial z|_{z=0}$ is formed with reverse-mode automatic differentiation (`torch.autograd.functional.jacobian`) and treated as a constant for the inference and spectral diagnostics. The optimised reference code $z^\star$ is computed by converged L-BFGS from the same origin, with the convergence protocol and diagnostics given in Section B.1; its per-decoder inference-scale robustness is reported in Section D.

Metrics and norms. Reconstruction error is mean squared error over all output elements. The primary metric is the reconstruction-space inference gap $\Delta_{\mathrm{MSE}} = \mathrm{MSE}_{z_1} - \mathrm{MSE}_{z^\star}$ equation 7. Spectral quantities are computed from the singular values σ_i of J_0: the Gram condition number $\kappa(M) = \sigma_1^2 / \sigma_{n_z}^2$ with $M = J_0^\top J_0$, and the participation ratio $\mathrm{pr}(J_0) = (\sum_i \sigma_i)^2 / \sum_i \sigma_i^2$ (Eq. 14). Manifold alignment is the per-principal-angle $\cos^2 \theta_i$ between $\mathrm{col}(J_0)$ and the top-n_z PCA subspace of 2,000 test images. All comparative figures report the mean and ± 1 standard deviation over the three seeds (Section B.2).

B.1 Computing and validating the reference code

Because $\mathrm{MSE}_{z^\star}$ anchors every comparison in this paper, we specify how the reference code $z^\star$ in equation 5 is computed and how its convergence is verified. With the decoder parameters fixed, we minimise $\frac{1}{2}\sum_i \|F_\theta(z_i) - x_i\|^2$ from the same origin $z_0 = 0$ used by the one-step rule, using L-BFGS with strong-Wolfe line search (history size 100, unit initial step). Optimisation runs in closure blocks and is governed by two stopping signals: a *loss-plateau* signal, which fires when the relative

range of the objective over the last ten closure evaluations falls below 10^{-7}, and a *gradient* signal, which fires when the per-element gradient norm $\|\nabla_z \mathcal{L}\|_2/\sqrt{n\, n_z}$ falls below 10^{-4}. A reference code is declared *converged* only when *both* signals fire; a hard cap of 1000 closures ($n_z \leq 32$) or 2000 closures ($n_z = 128$) bounds the budget, and codes that reach the cap are flagged rather than discarded. We adopt the dual criterion deliberately: in high latent dimensions the decoder landscape contains flat or weakly constrained directions (Section **??**), so the loss can plateau while the gradient remains non-negligible along a poorly identified ridge. Treating plateau alone as convergence would overstate how tightly $z^\star$ is pinned down.

One-step inference convention. The one-step code in equation 3 is evaluated in the exact form the GON is trained to use. For the squared loss written as a per-sample sum reduced by a mean over the batch, the origin gradient yields, for a decoder with spatial output of $H \times W$ pixels, $z_1 = \alpha\, J_0^\top\, (x - F_\theta(0))/(HW)$ for convolutional decoders and the corresponding unnormalised form for vectorised decoders; the constant is the training normalisation and is held identical across all decoders so that the comparison applies one fixed inference rule. We set the inference scale $\alpha = 1$ throughout, the value the GON is trained at, and treat α as the free scale of Section 4: since the scale-invariant quantity is the conditioning $\kappa(J_0^\top J_0)$ rather than the absolute scale, a per-decoder choice of α cannot create good one-step inference from a poorly conditioned decoder. We confirm this directly in Appendix D, where refitting α per decoder by a one-dimensional search leaves the family separation intact.

B.2 Seed variability and how it is reported

Every configuration is trained independently with three random seeds; the full architecture, training, and dataset protocol is deferred to Appendix B. The seed controls parameter initialisation and minibatch order, with data splits and all evaluation code—including the reference-code optimiser of Section B.1—held fixed across seeds, so each reported quantity is computed per seed and summarised across the three seeds of its configuration. This subsection fixes only the convention by which that across-seed variability is shown in the figures.

Throughout, a shaded region around a solid mean line denotes ± 1 standard deviation across the three seeds, and an error bar on a bar denotes the same; with only three seeds these dispersions are coarse estimates and we read them as such. Bar figures carry ± 1 standard-deviation error bars; the singular-value spectra are drawn as the across-seed mean with a ± 1 standard-deviation band shaded around it; and the per-angle alignment bars carry ± 1 standard-deviation error bars.

Application to the comparative figures. In Figure 1 the $\mathrm{MSE}_{z^\star}$ and Δ_{MSE} bars carry ± 1 standard-deviation error bars across seeds, so the GON–autoencoder gap separation is read against, not over, seed dispersion. In Figure 4 each panel shows the across-seed mean singular-value spectrum with a ± 1 standard-deviation band; the spectral flatness contrast is stable across seeds. In Figure 10 the per-angle alignment bars carry ± 1 standard-deviation error bars, but we use that figure as a descriptive diagnostic rather than as a controlled intervention. In Figure 5 each regularisation strength λ is plotted as the seed mean with a ± 1 standard-deviation band.

Is the non-monotone regulariser effect outside seed noise? Because the claim that stronger J_0-isometry regularisation *worsens* Δ_{MSE} on MNIST and FashionMNIST (Section 6) is a non-monotone effect, we test it against seed variability rather than asserting it from the mean curve. For each dataset we compare Δ_{MSE} at the small-λ and large-λ ends using the three paired seeds, and report the effect as established only where the per-seed differences are consistent in sign and the ± 1 standard-deviation bands at the two ends do not overlap. On the datasets where the bands do overlap we describe the trend as not separable from seed noise rather than as a worsening, and say so explicitly in the text and caption.

C $z^\star$ Convergence

Across the trained checkpoints, the strict dual criterion is met by a minority of cells, a larger group reaches the loss plateau without meeting the gradient tolerance, and the remainder reach the closure

Subset	GON n	GON mean Δ_{MSE}	AE n	AE mean Δ_{MSE}	AE/GON ratio
All (180 decoders)	90	0.0147	90	0.0733	4.98
Reliable (converged ∪ plateau)	53	0.0115	47	0.0759	6.63

Table 2: The GON–autoencoder inference-gap separation *strengthens* when weakly converged reference codes are dropped. The reliable subset retains every cell meeting the loss-plateau criterion (a trustworthy floor on the flat ridges of Section B.1) and discards only the cap cells, which are disproportionately under-converged autoencoders; dropping them *raises* the autoencoder-to-GON mean-gap ratio from 4.98 to 6.63. Means over decoders per family (5 datasets $\times$ 2 architectures $\times$ 3 latent dimensions $\times$ 3 seeds, minus flagged cells).

cap.[1] The cells that reach the cap without plateauing are concentrated at the largest latent dimension, and within that regime the autoencoder reference codes are systematically less converged than the GON ones (for example, at $n_z = 128$ the GON reaches a per-element gradient norm of order 10^{-6} while the matched autoencoder remains near 10^{-3} at the cap). We therefore report the central separation both on the full set and restricted to the converged-or-plateaued (*reliable*) subset (Table 2). This result does not weaken our results, in fact it *strengthens* them: the autoencoder-to-GON mean-gap ratio rises from 4.98 on all 180 decoders to 6.63 on the reliable subset, precisely because the discarded cap cells are disproportionately under-converged autoencoders whose gap is therefore *under*-estimated. We summarise on the reliable subset rather than the strict dual-criterion subset alone, since the latter retains too few cells per configuration (as few as 0–3 per family at some n_z) to estimate a stable ratio.

The convergence asymmetry is conservative for our claims. The direction of the bias is favourable. An under-converged reference *over*-estimates $\mathrm{MSE}_{z^\star}$, since optimisation stopped short of the decoder's true floor; by equation 7 this *reduces* the measured gap $\Delta_{\mathrm{MSE}} = \mathrm{MSE}_{z_1} - \mathrm{MSE}_{z^\star}$. Because the autoencoder reference codes are the less converged of the two families where they differ, the true autoencoder gap can only be *larger* than reported, and the true autoencoder capacity floor can only be *lower* (better) than reported. Both effects strengthen, rather than manufacture, our two empirical claims, i.e. that autoencoders retain a large one-step inference gap (Section 3) and that their optimised reconstruction is competitive with the GON.

The reported floors are the true floors. The cells the gradient criterion flags are nonetheless at the true reconstruction floor, not merely under-budgeted. Two checks confirm this. First, raising the L-BFGS budget tenfold (to $20{,}000$ closures) changes neither the loss nor the iterate on the flagged cells: the optimiser stalls at the same point because its line search, not its budget, terminates it. Second, and more tellingly, $20{,}000$ steps of Adam from the origin, a first-order optimiser with no line search to stall, never reaches a lower loss than the L-BFGS reference on these cells, and is sometimes markedly worse (on dSprites by 0.035). The flag therefore reflects a flat, ill-conditioned basin in which the gradient norm stays moderate while the loss has genuinely bottomed out; $\mathrm{MSE}_{z^\star}$ is the correct floor, and the diagnostics that rely on it, including the origin-anchor control of Section F, are sound even on the flagged datasets.

The separation is invariant to reference-code initialisation. As a final check that the gap is not an artefact of how $z^\star$ is found, we recompute every reference code from each family's own native code—the autoencoder from its encoder output, the GON from its one-step code—rather than from the origin. The reconstruction floor descends at or below the native code in every cell, and the GON–autoencoder gap is preserved and in fact strengthens: the autoencoder-to-GON mean-gap ratio rises from 4.98 to 5.6 on the full set, and to 12.5 on the reliable subset. The strict dual-criterion convergence yield does *not* improve under this warm start. Starting adjacent to a flat-ridge floor meets the loss-plateau criterion immediately while leaving a non-negligible gradient along the flat directions, which reinforces rather than weakens the flat-ridge picture above: $\mathrm{MSE}_{z^\star}$ is a stable

[1] In the sweep reported here, of 180 checkpoints 13 meet both criteria, 87 reach the loss plateau without the gradient tolerance, and 80 reach the hard cap. The plateau-without-gradient group corresponds to the flat-ridge regime above: $\mathrm{MSE}_{z^\star}$ is a reliable reconstruction floor there even though $z^\star$ itself is not uniquely pinned down.

Family	n	mean Δ_{MSE} (α=1)	mean Δ_{MSE} (best α)	mean $\alpha^\star$
GON	90	0.0147	0.0115	2.15
Autoencoder	90	0.0733	0.0544	5.61
AE/GON ratio		4.99	4.74	

Table 3: Per-decoder inference-scale robustness. Refitting the scalar step leaves the AE–GON gap separation essentially unchanged. Values are means over 90 decoders per family (5 datasets $\times$ 2 architectures $\times$ 3 latent dimensions $\times$ 3 seeds).

reconstruction floor and the family separation is independent of the reference-code initialisation, even though $z^\star$ itself is not uniquely identified.

D Inference-Scale Robustness

Throughout the paper the one-step code is evaluated at the inference scale $\alpha = 1$. A natural concern is that the AE's large gap is a scale artifact. The linear theory already says a scalar step cannot fix anisotropic conditioning, but we verify it directly in the nonlinear setting.

Per-decoder line search. Because the origin gradient is independent of α, the one-step code scales linearly, $z_1(\alpha) = \alpha J_0^\top (x - F(0))$. For each trained decoder we form the one-step direction once and minimise $\mathrm{MSE}(F(\alpha J_0^\top (x - F(0))), x)$ over 81 logarithmically spaced values $\alpha \in [0.1, 10]$.

Refitting α per decoder barely moves the separation (Table 3). The AE-to-GON mean-gap ratio is 4.99 at fixed $\alpha = 1$ and 4.74 at each decoder's best α. The AE frequently drives $\alpha^\star$ to the boundary of the search range and still cannot reach the GON regime, indicating that a poorly conditioned origin Jacobian cannot be rescued by rescaling the step.

E Latent Cosine as an Inference Diagnostic

A tempting alternative measures latent-space alignment between z_1 and $z^\star$,

$$\cos(z_1, z^\star) = \frac{z_1^\top z^\star}{\|z_1\| \|z^\star\|}.$$

This is not a reliable measure of inference quality. At high latent dimension, $z^\star$ need not be uniquely identified: decoder landscapes contain flat or weakly constrained directions, and moving through them can change the latent vector substantially while leaving the reconstruction nearly unchanged. In such cases latent cosine can collapse even when one-step inference remains excellent in reconstruction space.

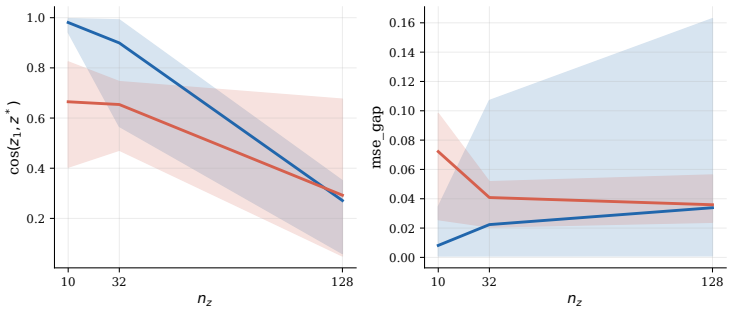


Figure 7: Latent-space cosine is an unreliable proxy for one-step inference quality. Left: cosine similarity between the one-step code z_1 and the optimised code $z^\star$ as latent dimension increases. Right: reconstruction-space inference gap for the same models. At high n_z the GON latent cosine can collapse even while Δ_{MSE} remains small, so latent alignment mismeasures the phenomenon.

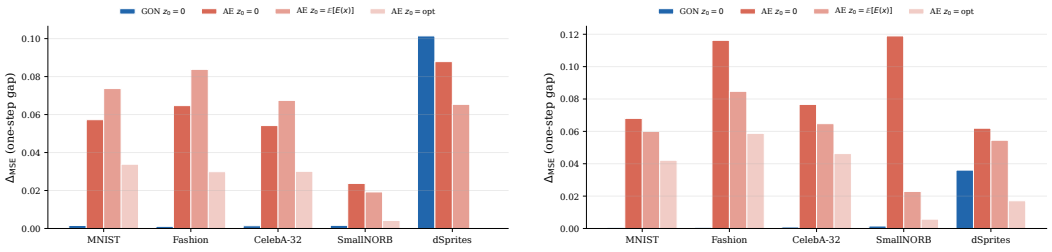
Figure 7 shows this directly: as n_z increases, the latent cosine between z_1 and $z^\star$ declines sharply, particularly for GONs, while Δ_{MSE} remains small. The latent metric treats movement in poorly identified or reconstruction-irrelevant directions as evidence against one-step inference, which it is not. We therefore use latent cosine only as a secondary diagnostic; all central claims are anchored to Δ_{MSE}, with $\mathrm{MSE}_{z^\star}$ distinguishing inference failure from decoder-capacity failure.

F Origin Anchor Control

A natural objection to anchoring one-step inference at $z_0 = 0$ is that zero may simply be the wrong origin for an autoencoder, whose own codes need not be centred there. We test this with *dataset-level* anchors; a single shared z_0, never tuned per example, and the generalised one-step rule

$$z_1 = z_0 + \alpha J_{z_0}^\top (x - F(z_0)), \qquad J_{z_0} = \left.\frac{\partial F}{\partial z}\right|_{z=z_0}.$$

For the AE we compare the origin $z_0 = 0$, its own mean code $z_0 = \mathbb{E}_x[E(x)]$, and an anchor optimised to minimise the dataset-mean one-step error. The final condition gives the AE a dataset-level fit the GON does not receive, so it is an upper bound on what moving the origin can buy.



(a) MLP (b) Convolutional

Figure 8: One-step gap under three shared origins for the autoencoder, against the GON at $z_0 = 0$. On natural-image datasets, even the best shared origin leaves the AE gap well above the GON's; on simpler datasets, especially dSprites, an optimised origin can nearly close the gap.

The result is not a clean null but something more informative. The autoencoder's own centre does not close the gap: moving the anchor to $\mathbb{E}_x[E(x)]$ leaves it unchanged or worse for the MLP, and reduces it only modestly for the convolutional decoder, never approaching the GON on the natural-image datasets. An origin optimised over the dataset is dataset-dependent. On MNIST, FashionMNIST, and CelebA-32 it roughly halves the gap but leaves it far above the GON's; on SmallNORB and dSprites it can bring the gap close to zero. Where the data manifold is simple enough that one shared anchor suffices, the origin location was indeed a major obstacle. On the natural-image datasets it is not, and the conditioning of the local map J_{z_0} remains the bottleneck no matter where the origin is placed. The independent $z^\star$ optimiser in Appendix B.1 confirms that these near-zero cases are true floors rather than reference-code artefacts.

G Linear Details

Proof of Proposition 1. Equality of the codes for a sample x requires $W^\top x = (W^\top W)^{-1} W^\top x$; left-multiplying by $W^\top W$ gives the per-sample condition. If $W^\top W = I$ it holds for every x. Conversely, if $z_1 = z^\star$ for every x, then $[(W^\top W)^{-1} - I] W^\top x = 0$ for all x; since W has full column rank, $W^\top$ has range $\mathbb{R}^{n_z}$, so $(W^\top W)^{-1} = I$ and $W^\top W = I$. For reconstruction, write the compact SVD $W = U \Sigma V^\top$ with $U^\top U = I$; then $P_W = UU^\top$ and $WW^\top = U\Sigma^2 U^\top$, so $WW^\top = P_W$ iff $\Sigma^2 = I$, equivalently $W^\top W = I$.

Proof of Proposition 2. For fixed W, the reconstruction WEx lies in $\mathrm{col}(W)$, whose best squared approximation to x is $P_W x$; this lower bound is achieved by $E^\star(W) = W^+ = (W^\top W)^{-1} W^\top$, since $WW^+ x = P_W x$. Taking expectations gives the stated minimum. For invertible A, $\mathrm{col}(WA) = \mathrm{col}(W)$, so $P_{WA} = P_W$ and the optimised loss is unchanged, while $(WA)^\top (WA) = A^\top (W^\top W) A$ generally changes the spectrum.

Step size. With a free inference scale α, the one-step code is $z_1 = \alpha W^\top x$, and $z_1 = z^\star$ for all x iff $W^\top W = \alpha^{-1} I$. The scale-invariant condition is therefore isotropy of the Gram matrix, $\kappa(W^\top W) = 1$.

Derivation of the decomposition equation 13. Any W of full column rank factors as $W = UA$ with $U \in \mathbb{R}^{D \times n_z}$, $U^\top U = I$, and $A \in \mathbb{R}^{n_z \times n_z}$ invertible; the QR decomposition of W supplies one such pair, and A collects the change of basis within $\mathrm{col}(W) = \mathrm{col}(U)$. Then $WW^\top = UAA^\top U^\top$ while $P_W = UU^\top$, so inserting $UU^\top x$ and regrouping,

$$x - WW^\top x = \underbrace{(I - UU^\top)x}_{\perp\, \mathrm{col}(U)} + \underbrace{U\big(I - AA^\top\big)U^\top x}_{\in\, \mathrm{col}(U)}. \tag{15}$$

The two terms are orthogonal, so their squared norms add. Because $U^\top U = I$, the map U is an isometry of $\mathbb{R}^{n_z}$ onto $\mathrm{col}(U)$ and the second norm simplifies, $\|U(I - AA^\top)U^\top x\| = \|(I - AA^\top)U^\top x\|$. Taking $\frac{1}{2}\mathbb{E}_x$ of equation 15 gives equation 13: the first term is the projection error of the subspace, which no choice of basis within it can change, and the second is the inference error contributed by the parametrisation alone.

The inference term vanishes precisely when $(I - AA^\top)U^\top x = 0$ almost surely. Provided the data is non-degenerate within the subspace, in the sense that $U^\top \mathbb{E}_x[xx^\top]U \succ 0$, this forces $AA^\top = I$; then $W^\top W = A^\top U^\top U A = A^\top A = I$, recovering the condition of Proposition 1. With a free inference scale the same argument gives $AA^\top = \alpha^{-1} I$, that is, a decoder basis orthonormal up to the inference scale.

Remark 1 (Residual symmetry). The GON objective is not invariant under $W \mapsto WA$ for general invertible A, but it does retain the orthogonal subgroup: for $Q^\top Q = I$ we have $(WQ)(WQ)^\top = WW^\top$, so $\mathcal{L}_{\mathrm{GON}}(WQ) = \mathcal{L}_{\mathrm{GON}}(W)$. The autoencoder's $GL(n_z)$ gauge freedom (Proposition 2) is therefore reduced to $O(n_z)$ by the one-step objective — the same reduction L_2 regularisation achieves on the linear autoencoder (Kunin et al., 2019).

Rank-deficient decoders. Propositions 1 and 2 assume W has full column rank, which is what makes $W^\top W$ invertible and $W^\top$ surjective. If W is rank deficient the statements carry over with inverses replaced by pseudoinverses: $z^\star = W^+ x$ is the minimum-norm least-squares code, $\hat{x}^\star = P_W x$ with $P_W = WW^+$ still the orthogonal projector onto $\mathrm{col}(W)$, and, writing the compact SVD $W = U\Sigma V^\top$ with $U^\top U = V^\top V = I$, one-step inference is exact for all x iff $\Sigma = I$, equivalently $W^\top W = VV^\top$, the orthogonal projector onto $\mathrm{row}(W)$ rather than the identity on all of $\mathbb{R}^{n_z}$. Note that the minimising code is no longer unique once W has a null space: any $z^\star + n$ with $n \in \mathrm{null}(W)$ attains the same reconstruction, so $z^\star$ is understood throughout as the minimum-norm solution $W^+ x$; since W annihilates $\mathrm{null}(W)$, the reconstruction statement $WW^\top x = P_W x$ is unaffected by that choice. We do not pursue this case further: every decoder in our sweeps has full column rank at the latent dimensions we report.

Linear detach ablation. The three conditions of Table 4 share the decoder $F_W(z) = Wz$ and the same squared reconstruction loss, and differ only in how z is produced and which occurrences of W receive gradient: the *linear GON* is tied, $z_1 = W^\top x$ and $\hat{x} = Wz_1$, with gradient through W in both places; the *linear AE* is untied, $z = W_e x$ and $\hat{x} = W_d z$; and the *linear detach* condition is tied in the forward pass only, $z_1 = \mathrm{sg}[W^\top x]$ with sg the stop-gradient, so gradient reaches the outer W alone. Each is trained on all five datasets at $n_z \in \{10, 32, 128\}$ with three seeds (135 runs) using Adam at learning rate 10^{-3}, batch size 512, 5000 steps, 60,000 training images and 500 held-out images for evaluation, with W initialised entrywise as $0.05 \cdot \mathcal{N}(0, 1)$.

The conclusion we draw is about Δ_{MSE}, which is near zero for both tied conditions and more than four orders of magnitude larger for the autoencoder at every latent dimension. We do not read the residual differences in $\kappa(W^\top W)$ between the GON and the detach condition as a property of the objectives. At this 5000-step budget the full GON has not yet reached isotropy on CelebA-32 for $n_z \geq 32$, which is the same under-convergence that motivates the longer 20,000-step runs used for CelebA-32 and SmallNORB in Figure 3; there the linear GON does reach $\kappa(W^\top W)$ close to one. The ablation is therefore evidence that forward tying alone is sufficient to remove the autoencoder-scale gap, not that detaching improves conditioning.

That forward tying suffices is itself the point, and it contradicts the natural prediction that removing the second-order term should push the model toward the autoencoder. It does not, because W still

appears in the reconstruction Wz_1: the loss sees the composite map $WW^\top$ and penalises non-isometry through the second term of equation 13 whether or not gradient flows through the inner $W^\top$. Section 6 shows the nonlinear case behaves differently, where $z_1 = J_0^\top(x - F(0))$ depends on the parameters through a Jacobian and the gradient pathway is necessary.

Condition	n_z	$\kappa(W^\top W)$	Δ_{MSE}
Linear GON	10	1.24 ± 0.29	$(3.0 \pm 4.1) \times 10^{-6}$
	32	2.18 ± 1.96	$(1.3 \pm 2.0) \times 10^{-5}$
	128	2.17 ± 1.68	$(9.3 \pm 14.0) \times 10^{-5}$
Linear detach	10	1.02 ± 0.03	$(1.0 \pm 0.9) \times 10^{-6}$
	32	1.13 ± 0.24	$(4.3 \pm 5.9) \times 10^{-6}$
	128	1.26 ± 0.49	$(1.6 \pm 2.7) \times 10^{-5}$
Linear AE	10	3.68 ± 1.31	7.05 ± 5.92
	32	5.15 ± 1.77	4.54 ± 2.77
	128	8.39 ± 2.27	1.85 ± 0.87

Table 4: Linear detach experiment: decoder Gram conditioning and the one-step inference gap for the full GON, the detach-GON, and the standard autoencoder. Each cell is the mean $\pm$ one standard deviation over the 15 runs at that latent dimension (5 datasets $\times$ 3 seeds, CelebA-32 and SmallNORB included).

H Excluded Datasets in the Linear Scatter

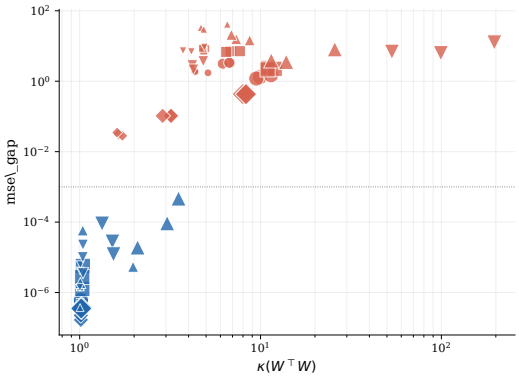


Figure 9: Linear-theory scatter for all five datasets, including dSprites and CelebA-32. As in Figure 3, the horizontal axis is $\kappa(W^\top W)$ and the vertical axis is Δ_{MSE}; linear GONs are blue, linear AEs red. The qualitative separation holds on all five datasets.

The clean main-text panel omits dSprites and CelebA-32 for readability. dSprites is synthetic and near-binary, so its AE points sit close to the GON cluster and compress the visual separation. CelebA-32 has very large and non-monotone AE gaps across latent dimension, obscuring the within-family trend. Neither pattern contradicts the theory: on both datasets, the linear GON achieves $\kappa(W^\top W)$ near one with near-zero gap, while the linear AE reaches a comparable reconstruction floor at $\kappa > 1$ with a much larger gap.

I Jacobian Metrics and Alignment Diagnostics

This appendix collects the secondary geometric diagnostics used to interpret Section 5. The singular-value diagnostics are central to the main argument, because the one-step update applies $J_0^\top$ and therefore uses the scale of each local direction. The alignment diagnostic is weaker: it asks whether $\mathrm{col}(J_0)$ overlaps a data subspace, but discards the gain attached to those directions. We therefore treat alignment as descriptive support, not as a controlled test of mechanism.

I.1 Spectral summaries of the origin Jacobian

Participation ratio. Let $\sigma_1 \geq \cdots \geq \sigma_{n_z} \geq 0$ be the singular values of J_0. We use $\mathrm{pr}(J_0) = (\sum_i \sigma_i)^2 / \sum_i \sigma_i^2$ because it is scale-invariant, equals n_z when the spectrum is flat, and equals 1 when J_0 is rank one. Stable rank and entropy-based effective rank give the same qualitative GON–AE comparison.

Active latent directions. For threshold $\tau \in (0, 1)$ define $\mathcal{A}_\tau(J_0) = \{i : \sigma_i \geq \tau\sigma_1\}$. "Near-isotropic over the active directions" means the condition number restricted to $i \in \mathcal{A}_\tau$ is close to one. We report $\tau = 10^{-2}$ and verify that the qualitative contrast is unchanged for $\tau \in [10^{-3}, 10^{-1}]$.

I.2 Manifold-alignment diagnostic

For each dataset we form a PCA basis V_{data} from 2,000 held-out images and compare it with the column space of the origin Jacobian. If Q_J is an orthonormal basis for $\mathrm{col}(J_0)$, the singular values of $Q_J^\top V_{\mathrm{data}}$ are $\cos\theta_i$, where θ_i are the principal angles between the two subspaces. Figure 10 plots $\cos^2\theta_i$ per angle.

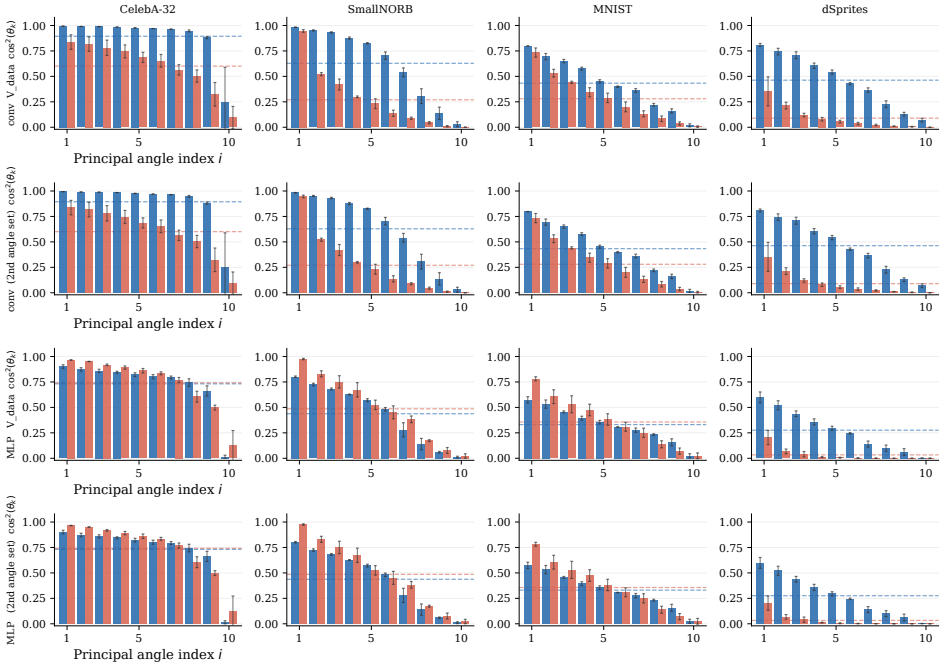


Figure 10: Per-angle alignment between the origin-Jacobian column space $\mathrm{col}(J_0)$ and a PCA approximation to the data manifold. This figure is a diagnostic rather than a mechanism test. Convolutional GONs often show stronger alignment than convolutional AEs, while the MLP comparison is more mixed and sometimes comparable. The important limitation is that principal angles ignore singular-value scale: a direction may be aligned with the data subspace but still have too little gain to support the one-step update.

The alignment results therefore narrow, rather than settle, the explanation. They show that J_0 is not an arbitrary local map unrelated to the data manifold, and that GON training can improve data-subspace alignment in some architectures. But they do not support a standalone alignment account of one-step inference. The one-step code depends on $J_0^\top (x - F(0))$, so raw subspace overlap must be read jointly with the singular-value spectrum of J_0 and with the self-consistency interventions in Section 6.

J INTERVENTION DETAILS

Each intervention removes one candidate explanation for the GON–autoencoder separation while holding the rest of the setting fixed: the same five datasets, the same decoder architectures, the same seeds $\{3, 4, 5\}$, and the same one-step evaluation rule of equation 3. This appendix gives the protocol for each, then reads Table 5 across datasets.

J_0-isometry regulariser. We add to the autoencoder's reconstruction loss a penalty that drives the origin Gram matrix toward a multiple of the identity, $\lambda \|J_0^\top J_0 - cI\|_F^2$ with $c = \mathrm{tr}(M)/n_z$. The penalty is estimated stochastically rather than by forming J_0: for random unit $v \in \mathbb{R}^{n_z}$ we have $\mathbb{E}\|J_0 v\|^2 = c$, so penalising the variance of $\|J_0 v\|^2$ over $n_{\mathrm{probe}} = 4$ Jacobian-vector products drives M toward cI, at the cost of four forward passes rather than n_z. The scale c is treated as a constant in the backward pass, so the penalty constrains the *conditioning* of M and not its overall gain. We sweep $\lambda \in \{0, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$ for 2500 steps on 60,000 training images, evaluating on 200 held-out images, at $n_z \in \{10, 32\}$ and three seeds. Table 5 reports the strongest setting, $\lambda = 1$; the full sweep is Figure 5.

Fresh decoder on frozen GON codes. This control supplies a decoder with the GON's codes while withholding the requirement that it generate them. In stage one we compute $z_1^{\mathrm{frozen}} = J_0^\top (x - F(0))$ from the trained GON over the training set and freeze them. In stage two we train a fresh decoder F', with the same architecture and initialisation scheme as the GON, to minimise $\mathrm{MSE}(F'(z_1^{\mathrm{frozen}}), x)$ with the codes held fixed, for 20,000 steps on 60,000 images. The decoder therefore solves a supervised regression in latent space with no coupling between what it reconstructs from and what its own Jacobian would produce. At evaluation F' is probed with *its own* origin Jacobian J_0', so $\kappa(M')$ and Δ_{MSE} measure whether exposure to good codes is by itself enough to organise the inference map.

Detach ablation. Because J_0 depends on decoder parameters, the nonlinear loss has a gradient pathway through the inferred code z_1 as well as through $F(z_1)$. Detaching z_1 during backpropagation keeps the forward inference rule intact but removes this pathway. We train both variants from scratch under otherwise identical settings: five datasets, $n_z \in \{10, 32\}$, three seeds, 10,000 steps, Adam at learning rate 5×10^{-4}, 200 evaluation images, and compare each detached run against its own full-GON baseline rather than against the autoencoder, since both variants are GON-trained. The ablation consistently worsens Δ_{MSE} and changes the learned geometry, establishing that the nonlinear gradient pathway is required. This contrasts with the linear case of Appendix G, where forward tying alone is sufficient.

Reading Table 5. On the four natural-image datasets the pattern is uniform. The GON reference gap is 0.001–0.003; both static interventions sit an order of magnitude or more above it, between $13\times$ and $115\times$ the GON gap, so neither imposed conditioning nor a supplied code distribution reaches the GON regime. The regulariser is the more erratic of the two: it worsens the gap relative to the plain autoencoder on MNIST, FashionMNIST and SmallNORB, and improves it substantially on CelebA-32 ($0.080 \to 0.015$) without closing it, which is the non-monotonicity documented in Figure 5 and tested against seed variability in Section B.2.

dSprites. dSprites' GON gap, 0.035, is an order of magnitude larger than on any other dataset, and the fresh-decoder control lands at 0.035 as well, equal to the GON reference rather than far above it. The cause is visible in the reconstruction floor rather than in the inference gap: at $n_z = 10$ an MLP decoder reaches $\mathrm{MSE}_{z^\star} \approx 0.20$ on dSprites for the GON and ≈ 0.24 for the autoencoder, against 0.011 and 0.009 on MNIST. Both families sit twenty to thirty times further from a good reconstruction than on the natural-image datasets, so the decoder is limited by what it can represent at this latent dimension rather than by how well the origin-gradient rule finds a code. This is exactly the regime the decomposition of Section 2 identifies as uninformative about inference: when $\mathrm{MSE}_{z^\star}$ is large the inference gap has little room in which to separate the families, and a control that inherits the same floor inherits the same gap. The dSprites cell therefore neither supports nor contradicts the claim that code distribution is insufficient; it is uninformative about it, and we exclude it when we describe the interventions as general. The same near-binary character excludes dSprites from the main-text linear scatter (Appendix H) and from the latent-cosine panel of Figure 7.

Dataset	GON	AE	AE + J_0-iso. reg.	Fresh decoder
MNIST	0.003	0.098	0.149	0.079
FashionMNIST	0.001	0.084	0.110	0.061
CelebA-32	0.001	0.080	0.015	0.053
SmallNORB	0.001	0.026	0.074	0.026
dSprites	0.035	0.076	0.178	0.035

Table 5: Generality of the static-property interventions across datasets (Δ_{MSE}, $n_z = 10$, MLP decoder, mean over three seeds). On every dataset the J_0-isometry regulariser and the fresh-decoder control leave Δ_{MSE} far above the GON reference: neither conditioning nor code distribution is sufficient. The regulariser ranges from only partly closing the gap (CelebA-32) to worsening it (MNIST, FashionMNIST, dSprites), never reaching the GON regime. The GON and AE columns are the reconstruction-task references of Figure 1; the intervention columns are from their respective experiments. dSprites is the near-binary outlier of Appendix H, where the GON one-step gap is itself large and the contrast is compressed. The detach ablation, a within-GON-training comparison, is reported against its own full-GON baseline in Section 6.

Seed variability is correspondingly large on dSprites: the per-seed GON gaps are $\{0.045, 0.041, 0.017\}$ and the autoencoder's $\{0.083, 0.094, 0.051\}$, standard deviations of roughly a third and a quarter of their means, against a few percent on MNIST. With three seeds the GON–autoencoder ordering on dSprites is still resolved, but the GON–fresh-decoder comparison is not.

K Related Work

Our work draws on five lines: amortised and optimisation-based inference; the linear-autoencoder loss landscape and its symmetries; sparse coding and frame theory; Jacobian conditioning and representation geometry; and implicit bias. We organise the discussion around the specific results each connects to, rather than by survey.

K.1 Review

Amortised, iterative, and encoder-free inference. Variational autoencoders established amortised inference as the standard solution for scalable latent-variable modelling (Kingma & Welling, 2014; Rezende et al., 2014). Subsequent work showed amortisation is itself a source of error: Cremer et al. (2018) decompose inference suboptimality into an approximation gap and an *amortisation gap*, and report that the divergence from the true posterior often reflects an imperfect recognition network and that the generator learns to accommodate the chosen approximation. This is the closest precedent for our reconstruction-space inference gap (Section 2), and the observation that the generator accommodates its inference scheme prefigures our finding that the trained GON decoder is organised around the codes produced by its own origin Jacobian (Section 6). Semi-amortised and iterative-amortised methods narrow the gap by refining encoder outputs with gradient-based updates (Kim et al., 2018; Marino et al., 2018); the one-step GON rule is the encoder-free, single-step limit of this idea. Generative Latent Optimization (Bojanowski et al., 2018) removes the encoder and optimises latents jointly with the decoder — the full-optimisation limit that our optimised reference code $z^\star$ instantiates with the decoder held fixed. Predictive coding relaxes latent activities by gradient descent on $\|x - G(z)\|^2$ plus a prior term (Rao & Ballard, 1999; Whittington & Bogacz, 2017; Millidge et al., 2022); for a linear decoder its inference operator is again the decoder transpose, its fixed point is the ridge-regularised least-squares code, and its first step from the origin is the matched filter — so GON is the one-step, prior-free truncation of predictive-coding inference. Marino (2022) names iterative-versus-amortised inference as the principal distinction between predictive coding and the VAE, which is the axis along which our GON–autoencoder comparison sits. Deep equilibrium models define a representation implicitly as a fixed point solved by root-finding and trained by implicit differentiation (Bai et al., 2019); our converged $z^\star$ is such a fixed point, and the implicit-differentiation view frames the detach experiment of Section 6, which removes the gradient through the inferred code that exact implicit differentiation would retain.

Linear autoencoders, PCA, and the symmetry the objective ignores. Classical results show that linear autoencoders trained with squared loss recover the principal subspace (Bourlard & Kamp, 1988; Baldi & Hornik, 1989), and later work recovers the ordered principal components from the learned weights post hoc, emphasising that the weights span — but need not equal — the loading vectors (Plaut, 2018). The loss-landscape analysis of Kunin et al. (2019) states the structure we rely on most directly: the unregularised objective is invariant under the action of $GL_k(\mathbb{R})$ given by $(E, W) \mapsto (GE, WG^{-1})$, its critical points satisfy $E = W^+$, and the optimal reconstruction is orthogonal projection onto $\mathrm{col}(W)$; L_2 regularisation reduces the symmetry to the orthogonal group, i.e. drives the decoder toward an orthonormal basis. Oftadeh et al. (2020) make the same invariance the centre of their analysis. We use these works to state plainly that our autoencoder-invariance result (Section 4) is a known property of the loss landscape; the addition is its inference reading — that this gauge freedom is exactly the freedom to leave $W^\top W$ ill-conditioned, which one-step inference cannot tolerate. The matched-filter perspective on this conditioning is old: Oja (1982)'s Hebbian rule extracts the principal subspace and renders the unit a matched filter for the data, and in frame theory the synthesis-adjoint (transpose) equals the canonical pseudoinverse exactly for a tight frame (Casazza & Kutyniok, 2013) — the $W^\top W = I$ condition of our Proposition 1.

Sparse coding, dictionary conditioning, and one-step decoding. Sparse coding infers a code through an inner optimisation against a fixed dictionary (Olshausen & Field, 1996), solved by iterative soft-thresholding (Daubechies et al., 2004; Beck & Teboulle, 2009). Its learned acceleration, LISTA, unrolls the solver into a network whose first iterate from a zero code is the thresholded matched filter $D^\top x$ — our z_1 up to the shrinkage induced by the sparsity prior (Gregor & LeCun, 2010). Moreau & Bruna (2017) show that this acceleration is governed by how nearly the dictionary Gram matrix can be diagonalised, with the gain concentrated in the early iterations and absent where the factorisation does not exist — the sparse-coding mirror of our Proposition 1 and of the conditioning intervention in Section 6. The recovery analogue is mutual coherence: one-step thresholding succeeds when the dictionary Gram matrix is near-identity off the diagonal (Donoho et al., 2006). We therefore position one-step GON inference as matched-filter decoding, exact under a tight-frame condition on the decoder, and we mark two differences. First, sparse recovery *assumes* an incoherent dictionary and asks whether cheap inference works; we ask the converse — which training objective *produces* the conditioning — an implicit-bias question rather than a recovery guarantee. Second, sparse coding has no untied encoder, and so no analogue of the autoencoder gauge freedom that our result turns on.

Jacobian conditioning and representation geometry. Contractive autoencoders penalise the Frobenius norm of the *encoder* Jacobian to learn locally invariant features (Rifai et al., 2011) — the encoder-side counterpart of the origin-Jacobian object we study — and regularised autoencoders have been analysed through the reconstruction Jacobian as estimators of the data score (Alain & Bengio, 2014). Dynamical isometry (Pennington et al., 2017) requires Jacobian singular values near one for stable signal propagation; we repurpose the same conditioning as an *inference* condition on J_0 (Section 5). A range of methods push weight spectra toward isometry — spectral normalisation (Miyato et al., 2018), stable-rank normalisation (Sanyal et al., 2020), and orthogonality or Parseval-frame regularisation (Saxe et al., 2014; Bansal et al., 2018; Brock et al., 2019; Cisse et al., 2017) — i.e. toward the $W^\top W \approx I$ condition that makes one-step inference exact. The pointed connection is to Section 6: explicitly regularising an autoencoder's J_0 toward isometry does *not* recover GON-level one-step inference. Our negative result thus qualifies the Jacobian-regularisation programme — well-conditioned J_0 is a signature of the trained GON state and the correct object by the linear theory, but it is not by itself the mechanism, because the autoencoder objective still does not organise good reconstructions around the origin-gradient codes.

Implicit bias, symmetry, and training-induced geometry. Deep linear networks exhibit structured learning dynamics not visible in the end-to-end objective (Saxe et al., 2014), and gradient descent selects particular solutions among an equivalence class (Soudry et al., 2018; Gunasekar et al., 2017). The continuous symmetry our result instantiates is exactly the kind studied here: the rescaling and reparametrisation symmetries of homogeneous and linear networks induce conserved quantities under gradient flow (Du et al., 2018; Kunin et al., 2021). GON training reliably selects a well-conditioned, high-participation-ratio J_0 even though the reconstruction loss does not penalise its conditioning; this is an implicit-bias story, and the autoencoder's blindness to conditioning is the

symmetry that breaks it. We adopt the discipline of the neural-collapse literature, where a striking geometric regularity is a robust feature of trained networks without, on its own, constituting a causal account of generalisation (Papyan et al., 2020; Hui et al., 2022): the near-isotropic J_0 spectrum is the correct object by the linear theory and a robust signature of the trained state, but the nonlinear weight-space mechanism that produces it remains only partially characterised (Section 7).

Identifiability and intervention-based analysis. That a representation is determined only up to reparametrisation is the general non-identifiability result of the disentanglement and nonlinear-ICA literatures (Locatello et al., 2019; Khemakhem et al., 2020); the autoencoder's GL_k gauge freedom is its linear instance, and GON breaks it by tying inference to the decoder transpose. Methodologically, we follow analyses that isolate load-bearing components by intervention rather than correlation. Our detach ablations parallel the stop-gradient analysis of SimSiam (Chen & He, 2021); our argument that latent cosine is an unreliable proxy for inference quality (Section **??**) aligns with the finding that representation-similarity verdicts depend on the chosen metric and its invariances (Kornblith et al., 2019); and the broader strategy of ruling out attractive explanations — here conditioning, code distribution, and alignment — follows work such as Tschannen et al. (2020) on mutual-information objectives and the dissection studies that perturb internal structure to test its causal role (Bau et al., 2017).

K.2 Implications

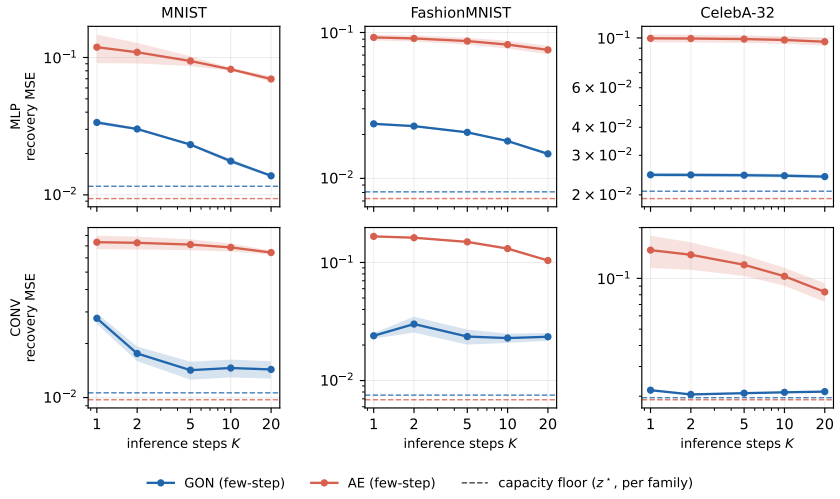


Figure 11: Trained decoders as priors for a compressed-sensing inverse problem ($n_z = 10$). Each decoder infers a latent from random linear measurements $y = \Phi x$ (with $m = D/4$ measurements) using the origin-gradient rule composed with Φ, from $z_0 = 0$; we plot full-image recovery MSE against the number of inference steps K (mean ± 1 SD over three seeds), with each family's full-optimisation capacity floor ($z^\star$ on the same task) shown dashed. GON decoders recover toward their capacity floor within a few steps, whereas autoencoder decoders stall far above their own—comparable—floor. The separation holds for both MLP (top) and convolutional (bottom) decoders across all three datasets; $n_z = 32$ is in Appendix L.

The mechanism has a reading beyond GONs. The autoencoder's blindness to conditioning is a *gauge freedom*: its objective is invariant to a group of decoder reparametrisations, and the property one-step inference needs lives inside that group. The general statement is that an inference scheme is blind to whatever a residual symmetry of the generative model leaves free; a learned encoder is maximally permissive, because it can absorb any invertible reparametrisation, whereas tying the inference rule to the decoder removes the freedom. This predicts where cheap test-time inference should and should not work, and we confirm the prediction directly. Used as priors in a linear inverse problem, trained decoders infer a latent from random compressed-sensing measurements $y = \Phi x$ ($m = D/4$) by the same origin-gradient rule, now composed with the measurement operator, from $z_0 = 0$. Under a fixed few-step budget the autoencoder decoders recover far worse than the GON

decoders—by factors of roughly 2–7 at the first step across three datasets, two architectures, and two latent dimensions—even though the two families have comparable optimised-code capacity on the same task, so the gap is inferential rather than a capacity effect (Figure 11). The deficit is therefore inherited by any downstream use of gradient-based latent inference: a decoder trained behind a flexible encoder is a poor few-step prior, because training never constrained the inference geometry. Our regularisation intervention shows the obvious remedy, a Jacobian penalty, does not reliably supply it; co-training the inference rule is the more dependable route.

This places the result within the wider family of gradient inference on a generative model. Across sparse coding, predictive coding, and GONs, cheap inference is competitive only when the generative model's local inference map is well conditioned, and whether the training objective *produces* that conditioning depends on whether inference is tied to the model: sparse coding imposes incoherence by design, while predictive coding and GONs tie inference to the generative map and can acquire it endogenously. The conditioning of the local inference map is thus a shared axis across the family, and whether an objective selects it without explicit regularisation—which this paper settles for GONs in the linear case and characterises empirically in the nonlinear one—is the discriminating question.

L Compressed-sensing Protocol

Section K.2 predicts that a decoder trained behind a flexible encoder should make a poor few-step prior for gradient-based latent inference. We test this on the same trained decoders, restricted to the three natural-image datasets (MNIST, FashionMNIST, CelebA-32), both architectures, and $n_z \in \{10, 32\}$, with the decoders *frozen*: no retraining, only inference.

Inverse problem and inference rule. For a linear measurement operator A we observe $y = Ax$ and recover a latent by iterating the origin-gradient rule composed with A, from $z_0 = 0$,

$$z_{t+1} = z_t + \alpha\, J_{z_t}^\top A^\top \big(y - AF(z_t)\big), \qquad t = 0, \dots, K-1,$$

which is exactly the one-step rule of Eq. 3 with the residual passed through $A^\top$; at $A = I$, $z_0 = 0$ it recovers z_1. The scale $\alpha = 1$ and the per-decoder training normalisation are held identical to the reconstruction experiments, so the same fixed rule is applied to every decoder. We use compressed sensing, $A = \Phi$ with $\Phi \in \mathbb{R}^{m \times D}$, $\Phi_{ij} \sim \mathcal{N}(0, 1/m)$ and $m = D/4$; the same Φ is used for both families in each setting. Recovery quality is the full-image MSE between $F(z_K)$ and the ground-truth x.

Capacity reference. As in the main text, a few-step failure is only informative if the decoder *can* solve the task. For each decoder we therefore also compute the full-optimisation floor $\min_z \|AF(z) - y\|^2$ (the dashed lines in Figure 11), the inverse-problem analogue of $\mathrm{MSE}_{z^\star}$. Across all settings the autoencoder capacity floor is comparable to, and often slightly below, the GON's (capacity ratios 0.8–1.0), so the few-step separation is an inference effect, not a capacity effect.

Why compressed sensing rather than inpainting. We also examined inpainting (observe a fixed pixel subset, recover the held-out region). There the recovery of the hidden pixels depends on the decoder's manifold, not only on the conditioning of the local inference map, so the observed-pixel optimisation floor is not a clean capacity ceiling for the held-out region: in several cells the GON's few-step recovery is better than its own observed-loss floor. Compressed sensing recovers the whole image from linear measurements and carries no such confound, so we report it as the quantitative result.

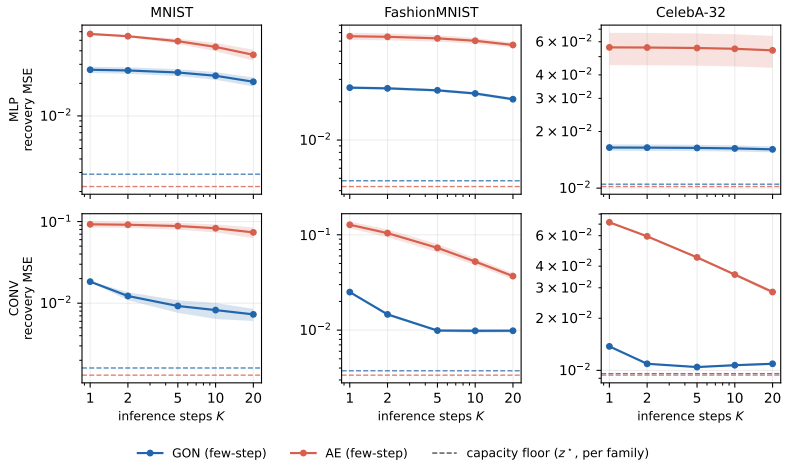


Figure 12: Compressed-sensing recovery as in Figure 11, for $n_z = 32$. The few-step GON–autoencoder separation, with comparable per-family capacity floors, is unchanged.