TOPOSTEER: TOPOLOGICAL LAYER SELECTION FOR ACTIVATION STEERING IN LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Activation steering controls large language models at inference time by adding a vector to their internal representations. Most work asks how to build that vector, and far less attention has gone to where to add it, even though steering at a poorly chosen layer can be worse than not steering at all. Prior selectors choose a layer without ever steering the model to test it: they score how separable the target behavior is at that layer, and how consistently the individual examples point along the steering direction. We argue that what matters is not the separability of the two activation clouds, nor the alignment of individual displacements with their mean, but the extent of the cloud of per-example displacements that mean-difference steering averages. We propose a topology-based layer selection method, called TOPOSTEER, which scores a layer by the total H_0 persistence of its displacement cloud, computed as the length of the cloud's Euclidean minimum spanning tree. We show that the score focuses on the displacement cloud's relative geometry, while class separation affects it only through standardization. We also provide a theoretical explanation for our method by proving that, as example-level variation increases, TOPOSTEER and discriminability move in opposite directions, so the two criteria can rank layers in opposite ways. Across multiple language models and steering tasks, TOPOSTEER provides a principled alternative to discriminability- and separability-based layer selection and yields more effective steering without requiring held-out validation.

1 INTRODUCTION

Large language models (LLMs) perform strongly across diverse tasks, yet real-world deployment increasingly demands controllable, reliable, and aligned behavior at inference time. While fine-tuning and reinforcement learning can improve alignment, they are costly, task-specific, and irreversible. Activation steering has therefore emerged as a lightweight alternative that controls LLMs behavior at inference time by intervening directly on internal representations, without modifying model parameters (Turner et al., 2023). The dominant recipe is mean-difference (MD) steering, which computes a steering vector from contrastive examples and injects it into selected layers, and which has shown effectiveness for AI alignment, model reasoning, and safety mitigation (Rimsky et al., 2024a; Nguyen & Le, 2026a;b). However, MD's steering effectiveness depends strongly on where the intervention is applied (Rimsky et al., 2024b; Sun et al., 2025).

This strong layer dependence makes validation-free layer selection an important practical problem: given the contrastive examples used to construct the steering vector, can we predict effective intervention layers without exhaustively evaluating steering at every candidate layer? Figure 1(a) illustrates the challenge using MD as an example. The effectiveness of MD steering varies substantially across injection layers: on Llama-8B-Instruct, steering at early layers leaves performance near the unsteered baseline, injecting at layer 15 improves it, and at other layers steering reduces performance below the baseline. Exhaustively selecting K of L layers requires evaluating $\binom{L}{K}$ possible layer subsets on held-out data, making such a brute-force search impractical. Recent work therefore scores layers using only the extraction data. LayerNavigator (Sun et al., 2025) ranks layers by $S_l = D_l + C_l$, combining a discriminability term D_l, defined as a Fisher-type ratio of between-class to total scatter along the steering direction, with C_l, which captures the alignment of individual displacements with the steering direction.

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

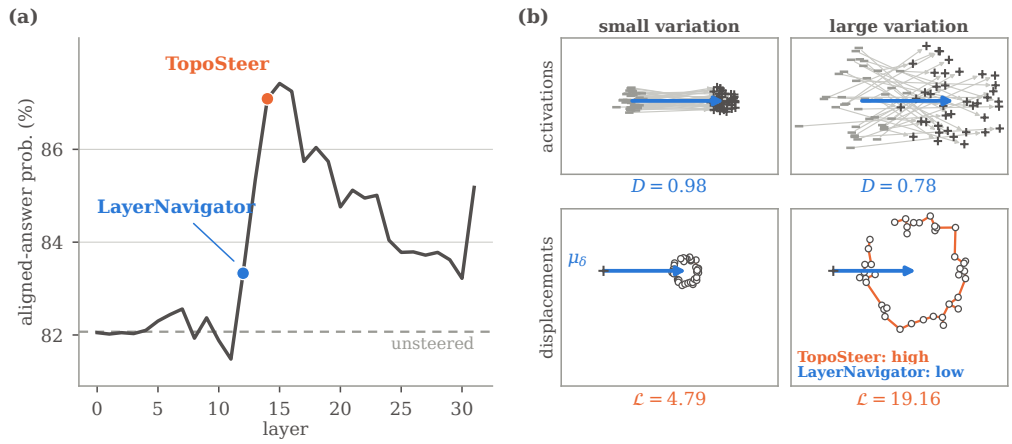

Figure 1: **Layer choice decides the outcome, and LayerNavigator and TopoSteer choose differently.** (a) Single-layer steering performance against depth (conscientiousness, Llama-8B) with the unsteered baseline, and the layer each method ranks highest (K=1). (b) Why they differ: two layers from the family of Theorem 1, with the same class means and the same steering vector, differing only in the scale of example-level variation. Top: activations, each contrastive pair joined by its displacement δ_l^i (grey arrows), with the mean difference μ_δ in blue. With large variation, the two classes overlap, so discriminability D drops, while the displacements keep a common direction but vary widely in length. Bottom: the same displacements drawn from a common origin form the displacement cloud Δ_l; connectivity length $\mathcal{L}$ is the length of its spanning tree (orange), which by Corollary 1 does not see the offset μ_δ at all. The large-variation layer is therefore scored higher by TopoSteer and lower by LayerNavigator, whose full score $D + C$ also drops. All values shown are computed from the data drawn; the construction is the one of Appendix F.

The two terms capture complementary but incomplete aspects of the geometry relevant to MD steering. MD does not use the two activation clouds separately. At layer l, let $a_l(x) \in \mathbb{R}^d$ denote the residual-stream activation of input x. We define the per-example displacement as $\delta_l^i := a_l(x_i^+) - a_l(x_i^-)$ (in the standardized representation of Sec. 3.1), and the displacement cloud as $\Delta_l := \{\delta_l^i\}_{i=1}^N$. Because MD steering forms its intervention by averaging these per-example displacements, whether a single vector represents the available signal is a property of the displacement cloud Δ_l. The discriminability term D_l never looks at that cloud: it is built from class means and within-class scatter, and describes how separable the two classes are rather than how the individual examples move. The consistency term does look at it, but only at directions; as an average of cosine similarities, it is unchanged if each displacement is rescaled independently. It therefore measures how tightly the displacements concentrate about the mean direction, and discards their extent: whether the cloud is a tight clump around a single vector, or a broad set of example-specific ones. This extent, overlooked by both terms, is what we propose to score in this paper.

To this end, we propose TopoSteer, which scores each layer l by the total degree-zero persistence of its displacement cloud

$$s_l \;:=\; \mathrm{TotPers}_0(\Delta_l) \;=\; \sum_{e \in \mathrm{MST}(\Delta_l)} w(e) \;=:\; \mathcal{L}(\Delta_l), \tag{1}$$

evaluated in the standardized representation of Sec. 3.1. The second equality is the classical identification of finite H_0 death scales with Euclidean minimum-spanning-tree edge weights (Gower & Ross, 1969), which we use as the computational form of the score. The score requires no held-out steering runs, no auxiliary training, and no forward passes beyond those already performed to extract the MD's steering vectors.

Our analysis shows that TopoSteer measures a quantity neither existing term captures. Against discriminability, the relation is an exact reversal: along the axis that scales example-level variation at fixed class means, the two scores move in opposite directions, so the criteria can rank layers oppositely (Theorem 1). Against consistency, it is complementarity: connectivity length is invariant to the mean displacement and 1-homogeneous, while consistency is an average of cosines and therefore

reads only directions, discarding the extent that $\mathcal{L}$ measures. On real activations, we further identify an *answer-polarity* axis induced by the contrastive format rather than the target behavior. This axis splits the displacement cloud into two lobes without contributing between-class information, yet it affects the two criteria through different geometric mechanisms: its effect on discriminability depends on its alignment with the steering direction, whereas its effect on $\mathcal{L}$ is governed by the spanning-tree bridge and the merging of the two lobes. We analyze these effects in Sec. 5.3.

Our main contributions are as follows. We identify the displacement cloud as the steering-relevant object for MD layer selection and propose TOPOSTEER, a validation-free topological selector based on its total H_0 persistence. We theoretically characterize how its connectivity differs from discriminability and directional consistency, and empirically show that it improves layer selection across multiple models and steering tasks without held-out search.

2 RELATED WORK

Activation Steering and Layer Selection. Activation steering computes a direction associated with a concept or behavior and adds it to residual-stream representations at inference time (Zou et al., 2023; Turner et al., 2023). A common approach constructs this direction from differences between paired contrastive activations, including contrastive activation addition (Rimsky et al., 2024b), refusal-direction interventions (Arditi et al., 2024), and related inference-time interventions (Li et al., 2023); the model parameters remain unchanged and the intervention can be applied dynamically (Nguyen & Le, 2026a;b). Our contribution is orthogonal to the choice of direction: we take the extractor as given and ask where to apply it. Existing selection strategies rely on probing accuracy, linear separability, activation differences, or heuristics such as intervening in the middle third of the network. LayerNavigator (Sun et al., 2025) is the closest point of comparison, combining a Fisher-type discriminability score D, computed from class means and within-class scatter, with a directional-consistency term C, the mean cosine between the steering vector and the individual displacements. TOPOSTEER scores the extent of those displacements, which neither term captures. TOPOSTEER evaluates the geometry of the per-example displacement vectors that MD steering actually averages, rather than the separability of the two activation classes.

Persistent Homology of Neural Representations. Persistent homology gives a multiscale description of a point cloud (Edelsbrunner & Harer, 2008), and has been used to study neural representations and their geometry (Rieck et al., 2019; Gabrielsson et al., 2020; Naitzat et al., 2020; Uchendu & Le, 2026). Fay et al. (2026) analyze LLMs' activation spaces and show that adversarial inputs induce topological compression, in which diverse small-scale features collapse into fewer dominant large-scale structures. Related work on transformer geometry has also documented systematic layerwise variation in intrinsic dimension, anisotropy, and representational expansion and compression with depth (Gardinazzi et al., 2025). Recent work has moved toward adaptive steering, selecting intervention layers or steering strengths based on the input. W2S (Gadgil et al., 2026) learns a predictor for the input-specific optimal intervention layer, using candidate-layer evaluations to construct the supervision signal. Dynamically Scaled Activation Steering (Ferrando et al., 2025) instead learns input- and layer-dependent scaling factors that determine when and how strongly to steer. Nelwan & Wicaksono (2026) extend adaptive selection to multiple layers, training a prompt-based ranker from per-instance layer-selection targets and an adaptive gate for the number of layers. These approaches require either candidate-layer evaluation or additional training to learn an input-dependent selection policy, whereas TOPOSTEER selects layers directly from the contrastive extraction data, without held-out steering evaluations or selector training. We therefore use persistent homology not as a classifier or training objective, but as a validation-free diagnostic of the geometry of paired displacements for selecting intervention layers.

3 TOPOSTEER: TOPOLOGICAL LAYER SELECTION

This section defines the method: the object it scores (Sec. 3.1), the score and its exact invariances (Sec. 3.2), and the selection procedure, its cost, the diagnostic that guards its main failure mode, and the two regimes in which the selected layers are steered (Sec. 3.3). Why this criterion differs from class-separability selectors, and what the displacement cloud looks like in a trained model, are deferred to Sec. 5, after the empirical comparison.

3.1 THE DISPLACEMENT CLOUD

Let l be a fixed layer of width d and let $\{(x_i^+, x_i^-)\}_{i=1}^N$ be paired contrastive inputs. Write $a_l(x_i^c) \in \mathbb{R}^d$, $c \in \{+,-\}$, for the residual-stream activation at the final prompt token. MD vectors are computed from the raw activations as follows: $v_l^{\text{raw}} := \frac{1}{N}\sum_{i=1}^N \big[a_l(x_i^+) - a_l(x_i^-)\big]$ and injects αv_l^{raw} at the chosen layers. For selection, we need displacement geometries that are comparable across layers rather than dominated by coordinate-wise scale, we standardize the representation with

$$\tilde{a}_l(x_i^c) = \frac{a_l(x_i^c) - \mu_l}{\sigma_l}, \tag{2}$$

where $\mu_l, \sigma_l \in \mathbb{R}^d$ are the coordinate-wise mean and standard deviation over all $2N$ activations at layer l. The *displacement cloud* is

$$\delta_l^i := \tilde{a}_l(x_i^+) - \tilde{a}_l(x_i^-), \qquad \Delta_l := \{\delta_l^i\}_{i=1}^N, \tag{3}$$

with mean $\bar{\delta}_l = \tilde{\mu}_l^+ - \tilde{\mu}_l^- =: \mu_{\delta,l}$ is the mean-difference direction in the analyzed space. Writing $\varepsilon_l(x_i^c) := \tilde{a}_l(x_i^c) - \tilde{\mu}_l^c$ for the within-class residuals and $\xi_l^i := \varepsilon_l(x_i^+) - \varepsilon_l(x_i^-)$ for the residual displacements, each displacement decomposes as

$$\delta_l^i = \mu_{\delta,l} + \xi_l^i, \qquad \sum_{i=1}^N \xi_l^i = 0, \qquad \Xi_l := \{\xi_l^i\}_{i=1}^N. \tag{4}$$

The contrastive construction gives Δ_l a second, structural decomposition that the score turns out to be insensitive to; we defer it to Sec. 5.3, where it is what separates the two criteria.

3.2 SCORING A LAYER

We score a layer by the connectivity of its displacement cloud, measured by the total degree-zero persistence $\mathrm{TotPers}(X)$ of a finite point cloud X, the sum of the finite death scales of its Vietoris–Rips H_0 barcode (Appendix B). Under the Euclidean filtration, these death scales are the edge weights of the Euclidean minimum spanning tree (Gower & Ross, 1969), which we use as the computational form of the score,

$$\mathrm{TotPers}(X) = \sum_{e \in \mathrm{MST}(X)} w(e) =: \mathcal{L}(X), \tag{5}$$

restated as Proposition 2 in Appendix C. It can be computed from the pairwise distance matrix in $O(n^2 d)$ time, and admits a direct interpretation as the connectivity length of the displacement cloud. TOPOSTEER assigns layer l the score:

$$s_l := \mathrm{TotPers}\left(\Delta_l\right) = \mathcal{L}(\Delta_l). \tag{6}$$

Two exact properties of $\mathcal{L}$ determine what the score can and cannot see (Proposition 4). It is translation invariant, and it is 1-homogeneous: $\mathcal{L}(X + u) = \mathcal{L}(X)$ for any $u \in \mathbb{R}^d$ and $\mathcal{L}(cX) = c\,\mathcal{L}(X)$ for $c > 0$. Applying translation invariance to the decomposition equation 4 gives the property we rely on throughout.

Corollary 1 (Common-mean invariance)**.** *Under the Euclidean metric, $\mathcal{L}(\Delta_l) = \mathcal{L}(\Delta_l - \mu_{\delta,l}) = \mathcal{L}(\Xi_l)$. The score is therefore invariant to the common mean displacement* of the cloud it is given*, and depends only on the relative arrangement of the example-level residual displacements. It does not follow that the pipeline is insensitive to between-class separation: the standardization equation 2 is estimated from the same activations, and separation enters through it. Proposition 1 makes the dependence exact.*

3.3 LAYER SELECTION AND STEERING PROTOCOLS

Layer Selection. Given a layer budget K, TOPOSTEER selects the K highest-scoring layers, S_K per task. At inference time, each selected layer receives its corresponding raw MD vector,

$$a'_{l,t} = a_{l,t} + \alpha v_l^{\text{raw}}, \qquad l \in S_K, \quad t \in \mathcal{T}, \tag{7}$$

where α is the steering strength and $\mathcal{T}$ denotes the intervention token positions. The standardized displacement cloud Δ_l is used only for layer scoring, while the raw MD vector v_l^{raw} is used for intervention; TOPOSTEER therefore changes only the layer-selection rule. Algorithm 1 summarizes the procedure. The same cached activations produce both the steering vectors and the layer scores, so TOPOSTEER requires no held-out steering evaluations, no auxiliary selector training, and no forward passes beyond those already needed for MD extraction; its additional cost is an offline geometric computation over the extraction data, $O(N^2 d)$ per layer (Appendix H). A large connectivity length can arise from edge weights spread across the cloud or from a few long edges bridging separated modes or isolated outliers; in the second case $\mathcal{L}$ measures separation between modes rather than the arrangement within them. Algorithm 1 therefore also returns the *bridge share*

$$b_l := \max_{e \in \mathrm{MST}(\Delta_l)} w(e) \,/\, \mathcal{L}(\Delta_l), \tag{8}$$

the fraction of the score carried by its longest edge, which flags whether the score is bulk-driven or carried by a single connection (Appendix F).

Simultaneous and Sequential Steering. We consider two intervention protocols. In the *simultaneous* regime, all K selected layers are intervened at once (equation 7) using a single extraction pass. It is the protocol used by prior validation-free selectors, and it assumes that scores computed on the unsteered model still describe the model once the first injection is in place, which may not be true. We therefore also evaluate a *sequential* regime, applying the K injections one at a time. At round k we re-run the extraction pass through the model already carrying rounds $1, \ldots, k-1$, recompute the MD vectors and the layer scores from those steered activations, and inject at the new argmax. A layer that remains the argmax is selected again and receives a second, re-estimated vector, so the output is an ordered multiset $(l^{(1)}, \ldots, l^{(K)})$ that need not equal the simultaneous top-K; at $K = 1$ the two regimes coincide. The regime costs K extraction passes instead of one and is a protocol rather than a selector.

Algorithm 1 TOPOSTEER Layer Selection

Require: $\mathcal{P} = \{(x_i^+, x_i^-)\}_{i=1}^N$; model M with L layers; steering budget K
Ensure: selected layers S_K; $\{v_l^{\text{raw}}\}_{l \in S_K}$; score $\{s_l\}_{l=1}^L$

```
Run M on P and cache {a_l(x_i^c)}_{l∈[L], i∈[N], c∈{+,−}}
for l = 1, ..., L do
    v_l^raw ← (1/N) Σ_{i=1}^N [a_l(x_i^+) − a_l(x_i^−)]
    (μ_l, σ_l) ← standardize {a_l(x_i^c)}_{i,c} by equation 2
    Construct Δ_l by equation 3
    T_l ← MST(Δ_l)
    s_l ← Σ_{e∈T_l} w(e)
    b_l ← max_{e∈T_l} w(e) / s_l
end for
S_K ← TopK_{l∈[L]} s_l
return S_K, {v_l^raw}_{l∈S_K}, {s_l}_{l=1}^L, {b_l}_{l∈S_K}
```

4 EXPERIMENTS

Setup. We evaluate TOPOSTEER on six binary persona-steering tasks from the Anthropic Persona Dataset (Perez et al., 2023). We use approximately 700 paired examples to extract the MD vectors and compute layer scores, and 100 held-out examples for evaluation. Experiments cover five instruction-tuned models: LLAMA-3-3B-INSTRUCT (Llama-3B), LLAMA-3-8B-INSTRUCT (Llama-8B) Grattafiori et al. (2024), QWEN2.5-3B-INSTRUCT (Qwen-3B), QWEN2.5-7B-INSTRUCT (Qwen-7B), and QWEN2.5-32B-INSTRUCT (Qwen-32B) Team et al. (2024). Activations are extracted at the final prompt token. TOPOSTEER scores the standardized displacement clouds of Eq. 3, while the raw MD vectors are used for intervention. Unless otherwise stated, $\alpha = 1.0$, $K \in \{1, 2, 3\}$, both intervention regimes in Sec. 3.3 are considered, and all filtrations use Euclidean distance. **Baselines.** We compare against LayerNavigator (Sun et al., 2025), layer selection by activation discriminativeness, random selection, and validation-based selectors requiring LN_{val} additional forward evaluations. We refer to Appendix I for additional detail.

Main Results. Table 1 summarizes alignment performance averaged over the six persona-steering tasks. In the simultaneous setting, TOPOSTEER outperforms LayerNavigator in 10 of 15 model–budget comparisons, ties in one, and is lower in four. The largest gains occur on Qwen-3B, where TOPOSTEER improves alignment from 64.30% to 71.77%, from 64.16% to 77.63%, and from 66.21% to 79.71% for $K = 1, 2, 3$, respectively. On Llama-8B, it also improves alignment at $K = 1$ and $K = 2$ by 2.18 and 1.22 percentage points, respectively. The gains are smaller and

		K = 1					K = 2					K = 3				
		Disc	Rand	LN	Ours	Ours$_{\ell_2}$	Disc	Rand	LN	Ours	Ours$_{\ell_2}$	Disc	Rand	LN	Ours	Ours$_{\ell_2}$
Simultaneous	LlamaB	70.72	69.96	**73.79**	72.79	72.79	79.71	75.40	**81.38**	**81.38**	**81.38**	77.03	73.44	75.12	**80.49**	77.40
	Llama-8B	79.73	80.04	81.26	**83.44**	82.70	81.70	83.83	84.87	84.26	**87.09**	84.11	85.79	**86.43**	85.91	84.73
	Qwen-3B	61.12	63.88	64.30	64.86	**71.77**	59.63	61.96	64.16	60.90	**77.63**	56.41	61.55	66.20	61.07	**79.71**
	Qwen-7B	**85.86**	84.91	84.80	85.84	85.72	85.85	85.86	**86.10**	85.85	85.62	85.63	85.19	84.97	85.88	85.40
	Qwen-32B	82.78	83.10	**83.42**	83.24	83.23	82.51	82.97	83.08	**83.91**	83.69	81.92	82.44	81.74	**83.66**	83.64
Sequential	LlamaB	70.72	69.96	**73.79**	72.79	72.79	78.74	73.20	**80.61**	80.41	80.41	77.12	76.01	75.17	**78.13**	73.06
	Llama-8B	79.73	80.04	81.26	**83.44**	82.70	81.14	81.49	84.28	84.11	**86.49**	82.61	83.10	85.24	85.58	**85.65**
	Qwen-3B	61.12	63.88	64.30	64.86	**71.77**	57.17	63.48	64.88	61.63	**76.72**	56.94	63.43	66.48	61.22	**79.19**
	Qwen-7B	**85.86**	84.91	84.80	85.84	85.72	81.34	85.40	**86.19**	85.78	85.51	80.28	**86.82**	86.30	85.66	85.38
	Qwen-32B	82.78	83.10	**83.42**	83.24	83.23	82.96	83.20	83.12	**83.53**	83.33	82.32	82.13	83.00	**83.33**	83.00

Table 1: Aligned-answer probability (%, ↑), averaged over the six persona-steering tasks. For each layer budget $K \in \{1, 2, 3\}$, we compare discriminability-only (Disc), random selection, LayerNavigator (LN), and TopoSteer under two scoring clouds: the standardized difference cloud (Ours) and its unit-sphere projection (Ours$_{\ell_2}$). **Bold** marks the best method per setting and underline the second best. Green cells highlight TopoSteer results that exceed the strongest non-TopoSteer baseline by at least 2.0 percentage point. The upper block reports simultaneous steering; the lower block reports greedy sequential selection with re-extraction and no layer repeats.

more mixed on Qwen-7B and Qwen-32B, showing that the benefit of displacement-based selection depends on both the model and layer budget. Per-task results and additional baselines are reported in Table 5. These results also reveal that increasing the number of selected layers does not necessarily improve alignment, motivating a separate analysis of the effect of the layer budget and its cost to output quality. We therefore report both alignment and generation perplexity in Table 6.

Sequential Steering. Table 1 compares simultaneous and greedy sequential steering at matched layer budgets and steering strength, with per-task results in Table 8. Sequential selection changes the results more noticeably than in our earlier aggregate comparison, but the relative ranking remains model-dependent. For Llama-3B, sequential steering favors LayerNavigator at $K = 2$ (80.61 vs. 80.41) but favors TopoSteer at $K = 3$ (78.13 vs. 75.17). On Llama-8B, TopoSteer slightly exceeds LayerNavigator at $K = 3$ (85.58 vs. 85.24). On Qwen-3B, the unit-sphere variant of TopoSteer remains substantially stronger at both $K = 2$ and $K = 3$ (76.72 and 79.19), while the standardized variant trails LayerNavigator. For Qwen-7B and Qwen-32B, the differences are smaller. Overall, sequential re-selection can change the preferred method as additional layers are introduced, but its effect depends strongly on both the model and the scoring variant.

Model Utility After Steering. Perplexity captures output fluency but does not directly measure whether steering preserves general model capabilities. We therefore evaluate the steered models on MMLU (Hendrycks et al., 2021). Figure 2 shows the change in MMLU accuracy relative to the unsteered model, averaged over $K = \{1, 2, 3\}$. Across models, the changes are small for all three steering variants, with TopoSteer remaining close to the unsteered baseline and showing near-zero average change on Qwen-7B and Qwen-32B. This suggests that the alignment gains from TopoSteer generally come with limited impact on downstream task performance. Full per-K results and comparisons are reported in Appendix M.



Figure 2: Change in MMLU accuracy relative to the unsteered model.

Which Layers TopoSteer Selects? Figure 3 illustrates how different selection criteria can lead to substantially different intervention layers. On Qwen-3B for `conscientiousness` at $K = 3$, Disc and LayerNavigator select layers 21–23, while TopoSteer$_{\ell_2}$ selects layers 33–35. This difference is accompanied by a large change in aligned-answer probability, from 51.58% for Layer-

	TopoSteer			W2S-Multi			
	Prob (↑)	PPL (↓)	Time (↓)	Prob (↑)	PPL (↓)	Time (↓)	ΔProb
Llama.2-3B	**78.22**	26.10	**7.7**	78.06	26.08	14.9	−0.16
Llama-8B	84.53	11.22	**10.7**	**85.83**	11.24	27.4	+1.30
Qwen-3B	62.28	7.00	**3.3**	**68.58**	8.44	6.1	+6.30
Qwen-7B	85.86	18.72	**4.6**	**89.03**	19.97	7.7	+3.17
Qwen-32B	83.60	5.84	**15.9**	**84.75**	6.92	23.6	+1.15
Average	78.90	13.78	**8.4**	**81.25**	14.53	15.9	+2.35

Table 2: Alignment probability (%) and perplexity averaged over six persona tasks and $K \in \{1, 2, 3\}$, and total running time in minutes for all six tasks for TopoSteer and W2S-Multi.

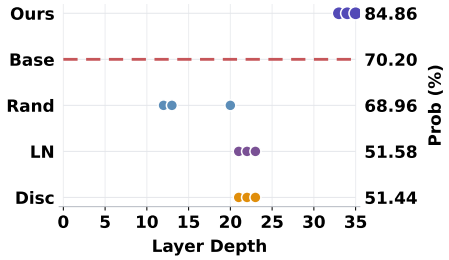

Figure 3: Layer Selection and Alignment Performance Across Methods.

Navigator to 84.86% for TopoSteer$_{\ell_2}$. The result illustrates that the choice of layer-selection criterion can matter as much as the number of layers selected. More generally, the two criteria rank layers according to different geometric properties of the representation, so their selected layer sets need not overlap. This case study provides an empirical view of the layer-selection shift; the full selected-layer sequences across models and tasks are reported in Appendix L.

Comparison with Evaluation-Based Selection. We additionally compare TopoSteer with W2S-Multi (Nelwan & Wicaksono, 2026), a per-instance multi-layer method that learns layer selection from evaluation-based steerability signals. Table 2 shows the alignment performance on the same five models across six persona tasks. Overall, W2S-Multi achieves higher average alignment (81.25% vs. 78.90%) at a slightly higher perplexity (14.53 vs. 13.78), but requires substantially more computation: 15.9 minutes versus 8.4 minutes for the complete evaluation. The additional cost arises from evaluating candidate layers on labelled validation prompts: its offline selection cost is $\mathcal{O}(LN_vC_f)$, where L is the number of candidate layers, N_v the number of validation prompts, and C_f the cost of a steered forward pass. In contrast, TopoSteer computes its layer score directly from the contrastive extraction data, without held-out steering evaluations or selector training. We provide setup and implementation details in Appendix N.

5 Connectivity versus Discriminability

The experiments show that connectivity often ranks layers better than separability in Table 1. We therefore ask why the two criteria differ: Sec. 5.1 identifies how separation affects the score; Sec. 5.2 shows that the criteria reverse their ordering along a controlled geometric axis; and Sec. 5.3 shows that they respond differently to the answer-polarity axis in real activations.

5.1 What Standardization Couples the Score To?

Corollary 1 says the score ignores the mean displacement of the cloud it is given. Separation still reaches it, through the standardization equation 2: for balanced classes the pooled variance of coordinate k is $\sigma^2_{l,k} = \bar{W}_{l,k} + \frac{1}{4}\big(\mu^{\mathrm{raw}}_{\delta,l,k}\big)^2$, where $\bar{W}_{l,k}$ is the pooled within-class variance of the raw activations and $\mu^{\mathrm{raw}}_{\delta,l}$ the raw mean difference.

Proposition 1 (Standardization couples the score to class separation)**.** *Write* $\rho_{l,k} := (\mu^{\mathrm{raw}}_{\delta,l,k})^2/(4\bar{W}_{l,k})$ *for the coordinate-wise separation-to-noise ratio, and* $H_l := \{\eta^i_l\}_{i=1}^N$ *with* $\eta^i_{l,k} := \xi^{\mathrm{raw},i}_{l,k}/\bar{W}^{1/2}_{l,k}$ *for the within-class-standardized residual displacements. Then*

$$\mathcal{L}(\Delta_l) \;=\; \mathcal{L}\big(H_l\Lambda_l\big), \qquad \Lambda_l := \mathrm{diag}\big((1+\rho_{l,k})^{-1/2}\big)_{k=1}^d,$$

so $\mathcal{L}(\Delta_l)$ *is non-increasing in every* $\rho_{l,k}$*, strictly decreasing in* $\rho_{l,k}$ *whenever coordinate* k *is non-constant on* Ξ_l*, and equal to* $\mathcal{L}(H_l)/\sqrt{1+\rho_l}$ *when the separation is spread evenly,* $\rho_{l,k} \equiv \rho_l$ *(proof in Appendix K).*

Separation therefore enters $\mathcal{L}$ only as a divisor. D_l is invariant to a common rescaling of all coordinates but not to the coordinate-wise one of equation 2, so both criteria move under the shared standardization, and part of any measured anti-correlation between them is attributable to it rather

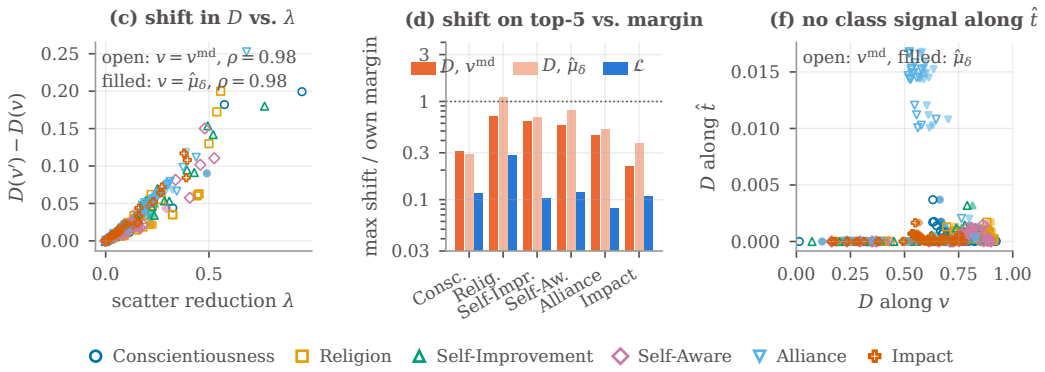

Figure 4: **Sensitivity to the answer-polarity axis**, Llama-8B, all six tasks and all 32 layers; panels keep the labels of the full Figure 5. (c) The shift $D(v') - D(v)$ against the scatter reduction λ of Proposition 5, with v the raw MD vector (open) or the standardized mean displacement $\hat{\mu}_\delta$ (filled); ρ is the Spearman correlation. (d) The largest shift among each criterion's own top-5 layers, divided by the gap between its rank-1 and rank-5 scores. (f) Discriminability along the polarity axis against discriminability along the steering direction.

than to displacement geometry. Appendix K gives the separation-free form $\mathcal{L}(H_l)$ of the score, which the same cached activations yield at no extra cost.

5.2 Decoupling from LayerNavigator's Discriminability

LayerNavigator scores a layer with a discriminability term D_l, a Fisher-type ratio of projected between-class to total scatter, together with a directional-consistency term. Consider the family that fixes the class means and scales the within-class residuals by $\sigma > 0$, giving displacements $\Delta_l(\sigma) = \{\mu_{\delta,l} + \sigma\xi_l^i\}_{i=1}^N$, with the standardization held fixed. Writing B_l and W_l for the projected between- and within-class terms, LayerNavigator's discriminability reduces to $D_l(\sigma) = B_l/(B_l + \sigma^2 W_l)$ (Appendix D).

Theorem 1 (Separability–connectivity decoupling)**.** *Assume $\mathcal{L}(\Xi_l) > 0$ and $B_l, W_l > 0$. Then, for every $\sigma > 0$, total H_0 persistence is strictly increasing in σ, whereas D_l is strictly decreasing. The two criteria therefore induce opposite orderings on the family $\{\Delta(\sigma)\}_{\sigma>0}$.*

Proof sketch. By Corollary 1 and 1-homogeneity, $\mathrm{TotPers}(\Delta(\sigma)) = \sigma\mathcal{L}(\Xi_l)$, so $\partial_\sigma \mathrm{TotPers} = \mathcal{L}(\Xi_l) > 0$. Differentiating $D_l(\sigma)$ gives $\partial_\sigma D_l = -2B_l\sigma W_l/(B_l + \sigma^2 W_l)^2 < 0$. The full proof and the reduction of $D_l(\sigma)$ are in Appendix D. □

The theorem concerns a one-parameter family at a fixed layer and the discriminability component alone: real layers may also differ along directions on which the two scores agree, which is why the selectors sometimes coincide, and the consistency component and the full published score enter through the comparison of Sec. 4. If the standardization is instead re-estimated along the family, $\mathcal{L}$ remains strictly increasing but saturates at the separation-free score $\mathcal{L}(H_l)$ of Proposition 1, and D_l keeps the reduction above whenever the re-standardization is a common scalar (Remark 1). A two-dimensional instance is Figure 1(b), constructed in Appendix F, which also shows the two ways a cloud can reach a high score, with length spread over many short edges or concentrated in a single bridge, the case the bridge share equation 8 detects.

5.3 Sensitivity to the Answer-Polarity Axis

The contrastive format imposes a second split on Δ_l, an *answer-polarity axis*, which we characterize and then remove, as a diagnostic, to see how each criterion responds. Formal statements and proofs are in Appendix E.

The answer-polarity axis. Equation 4 splits each displacement into a common part and a residual. The way contrastive pairs are built imposes a second split. Each example consists of one prompt and two candidate responses, one matching the target behavior and one not, and the activations $\tilde{a}_l(x_i^+)$

and $\tilde{a}_l(x_i^-)$ are taken after appending the matching and non-matching response, respectively. The prompt is the same in both, so it cancels in the difference, and what remains is the difference between the two response tokens plus a behavior term. In the persona datasets, the two responses are *Yes* and *No*. If the behavior-matching answer for example i is *Yes*, the token part of δ_l^i is $z_{\text{yes}} - z_{\text{no}}$; if it is *No*, the token part is the same vector with the opposite sign. Writing t_l for half that difference, every displacement carries either $+t_l$ or $-t_l$: $\delta_l^i = \pm t_l + b_l^i$ where b_l^i is the behavior term. The sign records how the dataset was written for that example, not anything about the behavior being steered. We estimate t_l as the half-difference of the displacement means of the *Yes*- and *No*-matching examples, and write $\hat{t}_l$ for the corresponding unit vector. Because the two polarities are balanced, the $\pm t_l$ contributions cancel in the mean, so MD steering is blind to the axis. The cloud is not: the two signs produce two lobes separated primarily along $\hat{t}_l$ (Figure 6). The axis carries essentially no class information: D_l evaluated along $\hat{t}_l$ is close to zero across tasks and layers (Figure 4(f)), as expected, since the sign records which response token is designated as behavior-matching, not the behavioral class. To isolate its effect, we project it out with $P_{\hat{t}_l} = I - \hat{t}_l\hat{t}_l^\top$; the projected cloud is a diagnostic only and is never used for steering.

Discriminability under projection. Let $v = \mu_{\delta,l}$ and $v' = P_{\hat{t}_l} v$, with unit vectors $\hat{v}$ and $\hat{v}'$. Projecting the activations and evaluating D_l along v' gives the same value as evaluating the original scatter matrices along v' (Lemma 3), so the projection acts on D_l as a change of direction, not of data. Let $\varepsilon := \langle \hat{v}, \hat{t}_l \rangle$ measure the alignment of the polarity axis with the steering direction, and $\lambda := 1 - W(\hat{v}')/W(\hat{v})$ the fractional reduction in within-class scatter from $\hat{v}$ to $\hat{v}'$. Then $B(\hat{v}') = (1 - \varepsilon^2)B(\hat{v})$ exactly, and

$$D(v') > D(v) \quad \Longleftrightarrow \quad \lambda > \varepsilon^2 \tag{9}$$

(Proposition 5): removing the axis improves discriminability only when the scatter it removes outweighs the between-class signal lost by tilting the steering direction. The shift vanishes when $\varepsilon = 0$, and when $\hat{t}_l$ is an eigenvector of S_l^w it is positive only if the axis carries more than $1+(1-\varepsilon^2)^{-1} \approx 2$ times the within-class scatter of the residual steering direction.

Connectivity under projection. For $\mathcal{L}$ the projection acts on the geometry itself. Let $\Delta_l' := P_{\hat{t}_l}\Delta_l$. Orthogonal projection cannot increase pairwise distances, so $\mathcal{L}(\Delta_l') \leq \mathcal{L}(\Delta_l)$, and the deficit is bounded by the components along $\hat{t}_l$ of the edges of the spanning tree of the projected cloud (Proposition 6). In a two-lobe cloud it has two parts: the bridge component along $\hat{t}_l$, and the superposition of the two lobes, which changes the tree from spanning two sets of about $N/2$ points to spanning N points in the residual space, a relative change of about $\log 2/D_{\text{int}}$ for lobes of intrinsic dimension D_{int} (Remark 2).

Comparison. The two criteria thus reach the projected-out direction through different mechanisms: $\mathcal{L}$ through its contribution to the spanning tree, D_l through the competition between ε and λ in equation 9. Figure 4 reports the measurements: panel (c) shows the shift in D against λ, and panel (d) puts both shifts on a common scale, dividing each by the gap between the criterion's own rank-1 and rank-5 layer scores. The projection is a diagnostic, not a fix: it does not reconcile the two rankings, moving LayerNavigator's ordering towards TOPOSTEER's on one task and away from it on the others. The claim is narrower: a criterion's value should not be dominated by a direction with zero between-class variance.

6 CONCLUSION

We introduced TOPOSTEER, a validation-free layer selector for mean-difference activation steering that uses the geometry of per-example displacement clouds rather than class separability. The score is total H_0 persistence, equivalently the Euclidean minimum spanning tree length, and requires neither held-out steering evaluations nor auxiliary training. Across six persona tasks and five models, TOPOSTEER improves alignment in many settings at comparable perplexity cost, supported by theoretical and geometric analyses contrasting connectivity and discriminability.

Limitations. The method assumes matched positive and negative examples and currently considers only mean-difference steering, leaving alternative extractors for future work. Multi-layer steering can degrade perplexity on some models, while Theorem 1 applies only to a fixed layer and its across-layer implications remain empirical; Appendix O further shows that the guarantees can miss mode splits concentrated in a few coordinates.

AI USE STATEMENT

In this work, we used generative AI tools for language editing and code assistance. Specifically, AI tools were used to correct grammar, improve clarity, polish manuscript text, and refine code for data visualization and related implementation tasks. We did not use generative AI tools to conduct experiments, generate or modify experimental results, construct datasets, or formulate scientific claims. All AI-assisted text and code were reviewed by the authors; generated code was manually inspected, executed, and verified for correctness. We take full responsibility for the final content of this work, including all text, claims, analyses, results, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Activation steering can be used to improve model behavior, such as increasing helpfulness or reducing undesirable responses, but it can also be misused to manipulate model outputs or bypass intended safeguards. This work studies controlled persona-style steering tasks and does not train or release a new model. Because the method operates at inference time, it could be applied dynamically without changing model weights. Any release of code or steering vectors should therefore include documentation of intended use, limitations, and potential misuse risks. Future deployment of steering methods should include safety evaluations to ensure that behavioral interventions do not degrade harmlessness, truthfulness, or robustness.

REPRODUCIBILITY STATEMENT

The score is defined in Sec. 3.2 and computed by Algorithm 1; all formal statements are proved in Appendices C–E with their assumptions stated. Tasks, models, baselines, metrics, steering strength, and layer budgets are given in Appendix I, per-task results in Appendix J, and the synthetic constructions behind every toy figure in Appendix F and Appendix O.

REFERENCES

Andy Arditi, Oscar Balcells Obeso, Aaquib Syed, Daniel Paleka, Nina Rimsky, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=pH3XAQME6c.

Herbert Edelsbrunner and John Harer. Persistent homology: A survey. *Contemporary Mathematics*, 2008. URL https://pub.ista.ac.at/~edels/Papers/2008-02-PersistentHomology.pdf.

Aideen Fay, Inés García-Redondo, Qiquan Wang, Haim Dubossarsky, and Anthea Monod. The shape of adversarial influence: Characterizing LLM latent spaces with persistent homology. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=v2PglvLLKT.

Alex Ferrando, Xavier Suau, Jordi Gonzàlez, and Pau Rodriguez. Dynamically scaled activation steering. *arXiv preprint arXiv:2512.03661*, 2025. URL https://arxiv.org/abs/2512.03661.

Rickard Brüel Gabrielsson, Bradley J. Nelson, Anjan Dwaraknath, and Primoz Skraba. A topology layer for machine learning. In Silvia Chiappa and Roberto Calandra (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 1553–1563. PMLR, 26–28 Aug 2020. URL https://proceedings.mlr.press/v108/gabrielsson20a.html.

Soham Gadgil, Chris Lin, and Su-In Lee. Where to steer: Input-dependent layer selection for steering improves llm alignment. *arXiv preprint arXiv:2604.03867*, 2026. URL https://arxiv.org/abs/2604.03867.

Yuri Gardinazzi, Karthik Viswanathan, Giada Panerai, Alessio ansuini, Alberto Cazzaniga, and Matteo Biagetti. Persistent topological features in large language models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=qAHnSkHvsm.

J. C. Gower and G. J. S. Ross. Minimum spanning trees and single linkage cluster analysis. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 18(1):54–64, 1969. ISSN 00359254, 14679876. URL http://www.jstor.org/stable/2346439.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv*, 2024. URL https://arxiv.org/abs/2407.21783.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in neural information processing systems*, 2023. URL https://openreview.net/forum?id=aLLuYpn83y.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=aajyHYjjsk.

Gregory Naitzat, Andrey Zhitnikov, and Lek-Heng Lim. Topology of deep neural networks. *Journal of Machine Learning Research*, 21(184):1–40, 2020. URL http://jmlr.org/papers/v21/20-345.html.

Muhammad Faishal Adly Nelwan and Alfan Farizki Wicaksono. Deployable per-instance multi-layer activation steering for large language models. *arXiv preprint arXiv:2608.08829*, 2026. URL https://arxiv.org/abs/2608.08829.

Tuc Nguyen and Thai Le. Atlas: Adaptive test-time latent steering with external verifiers for enhancing llms reasoning. *arXiv preprint arXiv:2601.03093*, 2026a. URL https://arxiv.org/abs/2601.03093v4.

Tuc Nguyen and Thai Le. Beyond linear activation steering: Invertible latent transformations for controlling llm behavior. *arXiv preprint arXiv:2606.08454*, 2026b. URL https://arxiv.org/abs/2606.08454.

Ethan Perez, Sam Ringer, Kamile Lukosiute, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, Andy Jones, Anna Chen, Benjamin Mann, Brian Israel, Bryan Seethor, Cameron McKinnon, Christopher Olah, Da Yan, Daniela Amodei, Dario Amodei, Dawn Drain, Dustin Li, Eli Tran-Johnson, Guro Khundadze, Jackson Kernion, James Landis, Jamie Kerr, Jared Mueller, Jeeyoon Hyun, Joshua Landau, Kamal Ndousse, Landon Goldberg, Liane Lovitt, Martin Lucas, Michael Sellitto, Miranda Zhang, Neerav Kingsland, Nelson Elhage, Nicholas Joseph, Noemi Mercado, Nova DasSarma, Oliver Rausch, Robin Larson, Sam McCandlish, Scott Johnston, Shauna Kravec, Sheer El Showk, Tamera Lanham, Timothy Telleen-Lawton, Tom Brown, Tom Henighan, Tristan Hume, Yuntao Bai, Zac Hatfield-Dodds, Jack Clark, Samuel R. Bowman, Amanda Askell, Roger Grosse, Danny Hernandez, Deep Ganguli, Evan Hubinger, Nicholas Schiefer, and Jared Kaplan. Discovering language model behaviors with model-written evaluations. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13387–13434, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.847. URL https://aclanthology.org/2023.findings-acl.847/.

Bastian Rieck, Matteo Togninalli, Christian Bock, Michael Moor, Max Horn, Thomas Gumbsch, and Karsten Borgwardt. Neural persistence: A complexity measure for deep neural networks using algebraic topology. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=ByxkijC5FQ.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, Bangkok, Thailand, August 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.828. URL https://aclanthology.org/2024.acl-long.828/.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, Bangkok, Thailand, August 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.828. URL https://aclanthology.org/2024.acl-long.828/.

Hao Sun, Huailiang Peng, Qiong Dai, Xu Bai, and Yanan Cao. Layernavigator: Finding promising intervention layers for efficient activation steering in large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=wj4lM45xQR.

Qwen Team et al. Qwen2 technical report. *arXiv*, 2024. URL https://arxiv.org/abs/2407.10671.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv*, 2023. URL https://arxiv.org/abs/2308.10248.

Adaku Uchendu and Thai Le. Topological data analysis applications in natural language processing: A survey. *SIGKDD Explor. Newsl.*, 28(1):81–101, June 2026. ISSN 1931-0145. doi: 10.1145/3820356.3820362. URL https://doi.org/10.1145/3820356.3820362.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, pp. 641–649, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400704314. doi: 10.1145/3626772.3657878. URL https://doi.org/10.1145/3626772.3657878.

Andy Zou, Long Phan, Alisa Liu, et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023. URL https://arxiv.org/abs/2310.01405.

CONTENTS

1 Introduction **1**

2 Related Work **3**

3 TOPOSTEER: Topological Layer Selection **3**

3.1 The Displacement Cloud 4

3.2 Scoring a Layer 4

3.3 Layer Selection and Steering Protocols 4

4 Experiments **5**

5 Connectivity versus Discriminability **7**

5.1 What Standardization Couples the Score To? 7

5.2 Decoupling from LayerNavigator's Discriminability 8

5.3 Sensitivity to the Answer-Polarity Axis 8

6 Conclusion **9**

A Notation **15**

B Topological Data Analysis **15**

B.1 Simplicial complexes 15

B.2 Vietoris-Rips filtration 15

B.3 Persistent Homology 15

C MST characterization via persistent homology **17**

D Derivation of the Discriminability Reduction **19**

E Formal Analysis of the Answer-Polarity Axis **21**

F Two-Dimensional Illustrations **26**

F.1 The decoupling theorem in 2D 26

F.2 Reading the MST of a displacement cloud 26

F.3 The lobe-merging term of Remark 2 28

G Why Mean-Difference Extraction **28**

H Computational Properties **28**

I Experimental Details **29**

I.1 Setup 29

J Detailed Alignment Results **30**

J.1 Alignment performance under total persistence score 30

J.2 Sequential Steering with Distinct Layers 30

K Separation-Free Standardization **30**

L Detail Layer Selection **30**

M Additional Analysis **30**

N Detail Performance Evaluation with Evaluation-Based Selection Method **31**

O Comparing Connectivity Length vs. Sum of Distance to the Mean **35**

O.1 Model 36

O.2 Lemma: sum of norms vs. connectivity length 36

O.3 Proof of Lemma 4 37

O.4 Two regimes 38

O.5 Simulation 39

O.6 Summary 40

A NOTATION

Table 3 collects the symbols used in the paper. A layer subscript l is carried where several layers are compared and suppressed where a single fixed layer is understood, so $\tilde{a}_i^c$ and $\tilde{a}_l(x_i^c)$ denote the same object. Four symbols carry more than one meaning; in each case the reading is fixed by context and is noted here to avoid ambiguity. The letter d denotes the residual-stream width in Sec. 5, a generic dissimilarity in the definition of the Vietoris–Rips complex, and a death scale when subscripted as d_i. The letter σ is the scale parameter of the homothety family, while σ_l is the vector of per-coordinate standard deviations used for Z-scoring. The letter α is the steering strength of Sec. 4. In Appendix O, which is self-contained, D denotes the ambient dimension of the synthetic clouds and N the number of points, rather than the discriminability and the number of pairs used in the main text. Finally, D_{int} in Remark 2 is the intrinsic dimension of a displacement lobe, not an ambient dimension. In Appendix E, B_l and W_l denote the quadratic forms evaluated at the *unit* steering direction $\hat{v}$; they differ from the B_l, W_l of Appendix D, evaluated at $\mu_{\delta,l}$ itself, by the factor $\|\mu_{\delta,l}\|^2$, which cancels in D_l.

B TOPOLOGICAL DATA ANALYSIS

Topological Data Analysis (TDA) provides a framework for studying the shape and structure of data by extracting topological features from discrete representations such as point clouds, graphs, or geometric datasets. In this work, we focus on point clouds obtained from neural network activations and use persistent homology to characterize their topological structure across different spatial scales.

B.1 SIMPLICIAL COMPLEXES

Simplicial complexes provide a convenient representation for studying the topology of discrete data. Given a set of vertices V, a simplicial complex K is a collection of non-empty subsets of V such that, whenever $\tau \in K$, all faces of τ are also contained in K. The elements of K are called simplices. This representation allows geometric information from a point cloud to be converted into a combinatorial structure on which topological properties can be computed.

B.2 VIETORIS-RIPS FILTRATION

Let (X, d_X) be a metric space, where in our setting d_X denotes the Euclidean distance. Given a scale parameter $\varepsilon \geq 0$, the Vietoris–Rips (VR) complex of a point cloud X is defined as

$$\mathrm{VR}_\varepsilon(X) = \{\emptyset \neq \tau \subseteq X \mid d(x_i, x_j) \leq \varepsilon,\ \forall x_i, x_j \in \tau\} \tag{10}$$

In other words, a simplex is included in the complex when all pairwise distances between its vertices are at most ε. As ε increases, more simplices are added, resulting in a nested sequence

$$\mathrm{VR}_{\varepsilon_1}(X) \subseteq \mathrm{VR}_{\varepsilon_2}(X), \qquad \varepsilon_1 \leq \varepsilon_2$$

This sequence is called the Vietoris–Rips filtration.

B.3 PERSISTENT HOMOLOGY

Once a filtration has been constructed, homology can be used to identify topological structures at each scale. In a simplicial complex, homology distinguishes structures such as connected components, loops, and higher-dimensional cavities.

For a simplicial complex K, the m-th homology group can be expressed as

$$H_m(K) = \frac{\ker \partial_m}{\mathrm{im}\, \partial_{m+1}},$$

where ∂_m is the boundary operator. Intuitively, this quotient identifies structures with no boundary that are not themselves boundaries of higher-dimensional structures. More specifically, H_0 captures connected components, H_1 captures one-dimensional loops, and H_2 captures enclosed voids.

Table 3: Notation Usage

Symbol	Meaning
Model and data	
l, L	layer index; number of layers
d	residual-stream width at the layer under consideration
N	number of contrastive pairs
x_i^+, x_i^-	the positive and negative variant of the i-th prompt
$a_l(x)$	residual-stream activation at layer l, final token position
v_l^{raw}	mean difference of raw activations
α, K	steering strength; number of layers intervened on
Normalized activations and scatter	
$\tilde{a}_i^c$	Z-scored activation of example i in class $c \in \{+, -\}$
μ_l, σ_l	coordinate-wise mean and standard deviation over all $2N$ activations
$\tilde{\mu}^c$	class mean of the normalized activations
$\varepsilon_l(x_i^c)$	within-class residual $\tilde{a}_i^c - \tilde{\mu}^c$
S^b, S^w	between-class and within-class scatter matrices, Eqs. 12 and 13
B, W	projected signal $\mu_\delta^\top S^b \mu_\delta$ and dispersion $\mu_\delta^\top S^w \mu_\delta$
D_l, $D_l(\sigma)$	LayerNavigator discriminability; its value on the family $(B/(B+\sigma^2 W))$
Displacements	
δ_i	displacement $\tilde{a}_i^+ - \tilde{a}_i^-$ of the i-th pair
Δ	displacement cloud $\{\delta_i\}_{i=1}^N$
μ_δ, $\hat{u}$	mean displacement $\bar{\delta}$ of the *standardized* cloud; its unit vector
ξ_l^i	residual displacement $\varepsilon_l(x_i^+) - \varepsilon_l(x_i^-)$, so $\delta_i = \mu_\delta + \xi_l^i$
Ξ	residual configuration $\{\xi_l^i\}_{i=1}^N$, i.e. Δ centered
$\Delta(\sigma)$	homothety family $\{\mu_\delta + \sigma\xi_l^i\}$, defined on the standardized cloud (Appendix D)
Topology	
$\text{VR}_\epsilon(X)$	Vietoris–Rips complex of X at scale ϵ
$\text{Dgm}(X)$, $B_0(X)$	persistence diagram; 0-dimensional barcode
d_i	death scale of the i-th H_0 class
$\text{TotPers}(X)$	total H_0 persistence $\sum_i d_i$ (essential class discarded)
$\text{MST}(X)$, $\mathcal{L}(X)$	minimum spanning tree; its total edge weight (connectivity length)
b_l	bridge share $\max_{e \in \text{MST}(\Delta)} w(e)/\mathcal{L}(\Delta)$, Eq. equation 8
W_1	1-Wasserstein distance on diagrams, ℓ^1 ground metric
Diag	diagonal $\{(t,t)\}$ of the persistence plane
$\hat{x}$, $P_{\hat{t}}$	unit vector $x/\|x\|$; orthogonal projector $I - \hat{t}\hat{t}^\top$

Persistent homology extends this analysis across the entire filtration. For two scales $\varepsilon_1 \leq \varepsilon_2$, the inclusion

$$\text{VR}_{\varepsilon_1}(X) \hookrightarrow \text{VR}_{\varepsilon_2}(X)$$

induces a map between their homology groups,

$$H_m(\text{VR}_{\varepsilon_1}(X)) \longrightarrow H_m(\text{VR}_{\varepsilon_2}(X)).$$

A topological feature can therefore be tracked as the scale parameter increases. Features that appear at some scale are said to be born, while features that subsequently disappear are said to die. Persistent homology records these birth and death scales, allowing topological features to be distinguished according to how long they persist across the filtration.

The resulting information can be represented by a barcode. For a fixed homological degree m, the barcode is a multiset of intervals

$$B_m(X) = \{[b_i, d_i)\}_{i=1}^{|B_m(X)|},$$

where each interval represents a topological feature, with b_i and d_i denoting its birth and death scales, respectively. The lifetime, or persistence, of the feature is $d_i - b_i$.

An alternative representation of the same information is given by persistence diagrams. For a fixed homological degree m, the persistence diagram $Dgm_m(X)$ is a locally finite multiset of points

$$\mathrm{Dgm}_m(X) = \{(b_i, d_i)\}_{i=1}^{|\mathrm{Dgm}_m(X)|} \subset \{(b,d) \in \mathbb{R} \times (\mathbb{R} \cup \{\infty\}) : b < d\},$$

together with the diagonal

$$\mathrm{Diag} = \{(t,t) : t \in \mathbb{R}\},$$

which is considered with infinite multiplicity. Each interval $[b_i, d_i)$ in the barcode is represented by a point (b_i, d_i) in the persistence diagram.

Finally, the total persistence in homological degree m is defined as the sum of the lifetimes of all features:

$$\mathrm{TotPers}_m(X) = \sum_{(b_i,d_i)\in B_m(X)} (d_i - b_i).$$

This quantity provides a compact measure of the overall amount of m-dimensional topological structure present in the point cloud across the considered range of scales. In degree 0, the sum is taken over the finite bars only: the single essential class, which never dies, is discarded, so $\mathrm{TotPers}_0$ is finite. We write $\mathrm{TotPers}$ for $\mathrm{TotPers}_0$ throughout.

To compare persistence diagrams, one possible metric is the Wasserstein distance. It carries two independent indices, which we keep apart: the order $q \in [1, \infty)$ of the sum, and the ground metric ℓ^r used on the persistence plane. For two diagrams,

$$W_q^{(r)}\big(\mathrm{Dgm}_m(X), \mathrm{Dgm}_m(X')\big) = \left(\inf_{\gamma} \sum_{p \in \mathrm{Dgm}_m(X) \cup \mathrm{Diag}} \|p - \gamma(p)\|_r^q \right)^{1/q},$$

where the infimum is taken over all admissible bijections

$$\gamma : \mathrm{Dgm}_m(X) \cup \mathrm{Diag} \longrightarrow \mathrm{Dgm}_m(X') \cup \mathrm{Diag}.$$

The use of the diagonal allows persistence features in one diagram to be matched to the diagonal, corresponding to features that are absent from the other diagram. Only the case $q = 1$ with the ℓ^1 ground metric is used in this paper; we abbreviate it $W_1 := W_1^{(1)}$, and it is the form appearing in Proposition 3(a).

C MST CHARACTERIZATION VIA PERSISTENT HOMOLOGY

A minimum spanning tree (MST) of a weighted graph is a spanning tree that connects all vertices while minimizing the total sum of edge weights. In the setting of a point cloud X, we consider the complete graph whose vertices are the points in X and whose edge weights are given by their pairwise Euclidean distances.

One standard way to construct an MST is Kruskal's algorithm. The algorithm considers the edges in non-decreasing order of their weights and adds an edge whenever it connects two previously disconnected components. Edges that would create a cycle are discarded. The process stops once all vertices are connected, resulting in an MST.

Throughout this section, we focus exclusively on connectivity and therefore on homology in degree 0. For simplicity of notation, we omit the homological degree from the notation for persistent homology quantities, with the convention that all such quantities refer to degree 0.

Proposition 2 (MST characterization)**.** *Let* $\mathrm{MST}(X)$ *be a minimum spanning tree of the complete weighted graph on* X*, with edge weights* $w(\{x_i, x_j\}) = \|x_i - x_j\|$*. The finite* H_0 *death scales are exactly the weights of the* $n-1$ MST *edges. Therefore,*

$$\mathrm{TotPers}(X) = \sum_{e \in \mathrm{MST}(X)} w(e) =: \mathcal{L}(X). \tag{11}$$

Proof. Let $B_0(X) = \{(0, d_i)\}_{i=1}^{n-1}$ denote the finite part of the 0-dimensional persistence diagram of the Vietoris-Rips filtration. In dimension 0, connected components of $\mathrm{VR}_\varepsilon(X)$ correspond precisely to the connected components of the graph $G_\varepsilon = (X, E_\varepsilon)$, where

$$E_\varepsilon = \{\{x_i, x_j\} \subset X \mid \|x_i - x_j\| \le \varepsilon\}.$$

A birth-death pair $(0, d) \in B_0(X)$ is created if and only if two previously disjoint connected components merge at scale $\varepsilon = d$. This merge is induced by the addition of an edge $\{x_i, x_j\}$ of length $\|x_i - x_j\|$. Edges whose addition creates cycles in G_ε do not alter the number of connected components and thus contribute no H_0 persistence classes.

By construction, this process mirrors the execution of Kruskal's algorithm on X: edges are inspected in non-decreasing order of weight, and an edge is added to the spanning forest if and only if it connects two distinct connected components. Consequently, the scale values $\{d_1, \dots, d_{n-1}\}$ at which H_0 classes die are precisely the weights of the $n - 1$ edges selected by Kruskal's algorithm to form $\mathrm{MST}(X)$.

$$\mathrm{TotPers}(X) \;=\; \sum_{i=1}^{n-1} d_i = \sum_{e \in \mathrm{MST}(X)} w(e) \;=:\; \mathcal{L}(X).$$

When some pairwise distances are tied, the minimum spanning tree need not be unique, but the multiset of its edge weights is; the identity is therefore unaffected by the tie-breaking rule. □

Proposition 3 (Stability)**.** *Let X and X' be finite subsets of $\mathbb{R}^d$ with $|X| = |X'| = n$. Then the following properties hold:*

(a) As a functional of the persistence diagram, TotPers *is W_1-Lipschitz (for an admissible bijection all but finitely many terms of the sum vanish). More precisely,*

$$|\mathrm{TotPers}(X) - \mathrm{TotPers}(X')| \le W_1(\mathrm{Dgm}(X), \mathrm{Dgm}(X'))$$

(b) As a functional of the point cloud, for any bijection $\phi : X \to X'$,

$$|\mathcal{L}(X) - \mathcal{L}(X')| \le 2(n-1) \max_i \|x_i - \phi(x_i)\|.$$

The constant grows linearly in n and is attained at $n = 2$; we make no claim of sharpness for larger n. The bound is therefore per-point and gives no control uniform in the number of points.

Proof. (a) We prove that TotPers is W_1-Lipschitz. Let

$$\gamma : \mathrm{Dgm}(X) \cup \mathrm{Diag} \longrightarrow \mathrm{Dgm}(X') \cup \mathrm{Diag}$$

be any admissible bijection, where $\mathrm{Diag} = \{(t, t) : t \ge 0\} \subset \mathbb{R}^2$ denotes the diagonal of the persistence plane. For a persistence point $p = (b, d)$, define its persistence by $\mathrm{Pers}(p) := d - b$.

For every matched pair $p = (b, d)$ and $q = (b', d')$, we have

$$|\,\mathrm{Pers}(p) - \mathrm{Pers}(q)| = |(d - d') - (b - b')| \le |d - d'| + |b - b'| = \|(b - b', d - d')\|_1 = \|p - q\|_1.$$

Therefore,

$$\begin{aligned} |\,\mathrm{TotPers}(X) - \mathrm{TotPers}(X')| &= \left| \sum_{p \in \mathrm{Dgm}(X) \cup \mathrm{Diag}} \big(\mathrm{Pers}(p) - \mathrm{Pers}(\gamma(p))\big) \right| \\ &\le \sum_{p \in \mathrm{Dgm}(X) \cup \mathrm{Diag}} |\,\mathrm{Pers}(p) - \mathrm{Pers}(\gamma(p))| \\ &\le \sum_{p \in \mathrm{Dgm}(X) \cup \mathrm{Diag}} \|p - \gamma(p)\|_1. \end{aligned}$$

Since γ was arbitrary, taking the infimum over all admissible bijections gives

$$|\,\mathrm{TotPers}(X) - \mathrm{TotPers}(X')| \le W_1(\mathrm{Dgm}(X), \mathrm{Dgm}(X'))\,.$$

(b) Let $\varepsilon = \max_i \|x_i - \phi(x_i)\|$. For every edge $e = \{x_i, x_j\}$ of $\mathrm{MST}(X)$, let $\phi(e) = \{\phi(x_i), \phi(x_j)\}$ be the corresponding edge in X'. By the triangle inequality,

$$\begin{aligned} w(\phi(e)) &= \|\phi(x_i) - \phi(x_j)\| \\ &\le \|\phi(x_i) - x_i\| + \|x_i - x_j\| + \|x_j - \phi(x_j)\| \\ &\le w(e) + 2\varepsilon. \end{aligned}$$

The collection of edges $\{\phi(e) : e \in \mathrm{MST}(X)\}$ forms a spanning tree of X'. Since $\mathrm{MST}(X')$ minimizes the total weight among all spanning trees of X',

$$\mathcal{L}(X') \le \sum_{e \in \mathrm{MST}(X)} w(\phi(e)) \le \sum_{e \in \mathrm{MST}(X)} \big(w(e) + 2\varepsilon\big) = \mathcal{L}(X) + 2(n-1)\varepsilon,$$

because every spanning tree has exactly $n - 1$ edges.

Applying the same argument to the inverse bijection $\phi^{-1} : X' \to X$ yields

$$\mathcal{L}(X) \le \mathcal{L}(X') + 2(n-1)\varepsilon.$$

Hence,

$$|\mathcal{L}(X) - \mathcal{L}(X')| \le 2(n-1)\varepsilon = 2(n-1) \max_i \|x_i - \phi(x_i)\|.$$

The constant is sharp: for $n = 2$, moving both points apart by ε each along the line through them lengthens the single edge by exactly $2\varepsilon = 2(n-1)\varepsilon$. □

Proposition 4 (Invariances). *For $t \in \mathbb{R}^d$ and $c > 0$: $\mathcal{L}(X + t) = \mathcal{L}(X)$ and $\mathcal{L}(cX) = c\,\mathcal{L}(X)$.*

Proof. For every edge $e = \{x_i, x_j\}$ of $\mathrm{MST}(X)$, $t \in \mathbb{R}^d$ and $c > 0$,

$$\|(x_i + t) - (x_j + t)\| = \|x_i - x_j\|, \qquad \|cx_i - cx_j\| = c\|x_i - x_j\|,$$

hence $\mathcal{L}(X + t) = \mathcal{L}(X)$ and $\mathcal{L}(c\,X) = c\,\mathcal{L}(X)$. □

D Derivation of the Discriminability Reduction

This appendix expands the algebra compressed in Sec. 5.2, in the notation established there.

We work at a fixed layer l in the notation of Sec. 3.1, writing $\tilde{a}_l(x_i^c)$ for the normalized activation of the i-th example under class label $c \in \{+, -\}$ and $\tilde{\mu}_l^c$ for the corresponding class mean. We define the residual of each sample with respect to the mean of its class as

$$\varepsilon_l(x_i^c) = \tilde{a}_l(x_i^c) - \tilde{\mu}_l^c,$$

where $c \in \{+, -\}$ and $\tilde{\mu}_l^c$ for class c. By construction,

$$\sum_{i=1}^{N} \varepsilon_l(x_i^c) = 0.$$

Instead of working with the $2N$ normalized activations, we consider the N paired differences. Specifically, we define

$$\delta_l(x_i) = \tilde{a}_l(x_i^+) - \tilde{a}_l(x_i^-) = (\tilde{\mu}_l^+ - \tilde{\mu}_l^-) + \big(\varepsilon_l(x_i^+) - \varepsilon_l(x_i^-)\big) := \mu_{\delta,l} + \xi_l^i,$$

where $\mu_{\delta,l} := \tilde{\mu}_l^+ - \tilde{\mu}_l^-$ and ξ_l^i denote the differences between the class means and between the class residuals, respectively.

We denote the centered residual difference ξ_l^i, and let $\Xi = \{\xi_l^i\}_{i=1}^N$ be the resulting cloud of residual differences. Since $\sum_i \varepsilon_l(x_i^c) = 0$ for each c, we have $\sum_{i=1}^N \xi_l^i = 0$: the residual configuration Ξ is precisely the displacement cloud Δ with its mean removed, so $\mathcal{L}(\Delta) = \mathcal{L}(\Xi)$ by Proposition 4.

To study the effect of varying the residual dispersion while keeping the class means fixed, we define the family directly on the standardized displacement cloud,

$$\Delta(\sigma) := \{\mu_{\delta,l} + \sigma\xi_l^i\}_{i=1}^N, \qquad \sigma > 0,$$

realized by the representative activations

$$\tilde{a}_l(\sigma)(x_i^c) := \tilde{\mu}_l^c + \sigma\, \varepsilon_l(x_i^c), \qquad c \in \{+,-\},$$

which keep the class means fixed and reproduce $\Delta(\sigma)$ exactly, since $\tilde{a}_l(\sigma)(x_i^+) - \tilde{a}_l(\sigma)(x_i^-) = \mu_{\delta,l} + \sigma\xi_l^i$. The standardization equation 2 is held fixed along the family — it is estimated once from the layer's own activations — so that S^b is unchanged and $S^w(\sigma) = \sigma^2 S^w$; Remark 1 treats the case in which it is re-estimated.

LayerNavigator's discriminability is

$$D_l(\sigma) = \frac{\mu_{\delta,l}^\top S_l^b \mu_{\delta,l}}{\mu_{\delta,l}^\top \left(S_l^b + S_l^w(\sigma)\right) \mu_{\delta,l}},$$

where S_l^b and S_l^w denote the between-class and within-class covariance matrices, respectively,

$$S_l^b = N \sum_{c\in\{+,-\}} (\tilde{\mu}_l^c - \tilde{\mu}_l)\,(\tilde{\mu}_l^c - \tilde{\mu}_l)^\top, \tag{12}$$

$$S_l^w(\sigma) = \sum_{c\in\{+,-\}} \sum_{i=1}^{N} (\tilde{a}_l(\sigma)(x_i^c) - \tilde{\mu}_l^c)\,(\tilde{a}_l(\sigma)(x_i^c) - \tilde{\mu}_l^c)^\top. \tag{13}$$

Using the notation introduced above, the expressions in equation 12 and equation 13 simplify to

$$S_l^b = \frac{N}{2} \mu_{\delta,l} (\mu_{\delta,l})^\top \tag{14}$$

$$S_l^w(\sigma) = \sum_{c\in\{+,-\}} \sum_{i=1}^{N} \sigma\varepsilon_l(x_i^c)\,(\sigma\varepsilon_l(x_i^c))^\top = \sigma^2 S_l^w. \tag{15}$$

We define $B_l(v) = v^\top S_l^b\, v$ and $W_l(v) = v^\top S_l^w\, v$ such that

$$B_l := \mu_{\delta,l}^\top S_l^b \mu_{\delta,l} = \frac{N}{2}\|\mu_{\delta,l}\|^4, \tag{16}$$

$$W_l := \mu_{\delta,l}^\top S_l^w \mu_{\delta,l} = \sum_{c\in\{+,-\}} \sum_{i=1}^{N} \left(\mu_{\delta,l}^\top \varepsilon_l(x_i^c)\right)^2. \tag{17}$$

The within-class term W_l is close to, but not equal to, the spread of the displacement cloud along the same direction, and the gap is exactly the paired covariance.

Lemma 1 (What W_l and the displacement spread differ by)**.** *With* $a_i := \mu_{\delta,l}^\top \varepsilon_l(x_i^+)$ *and* $b_i := \mu_{\delta,l}^\top \varepsilon_l(x_i^-)$,

$$\sum_{i=1}^{N} \left(\mu_{\delta,l}^\top \xi_l^i\right)^2 \;=\; W_l \;-\; 2\sum_{i=1}^{N} a_i b_i.$$

Proof. $\mu_{\delta,l}^\top \xi_l^i = a_i - b_i$, so $\sum_i (a_i - b_i)^2 = \sum_i a_i^2 + \sum_i b_i^2 - 2\sum_i a_i b_i$, and the first two sums are W_l by its definition above. □

Variation shared by the two members of a pair — prompt template, topic, formatting, therefore enters W_l but cancels in ξ_l^i. This is the structural reason the two criteria can come apart, and it is what the shared term c_i isolates in the construction of Appendix F.

Substituting these expressions into the definition of the LayerNavigator's discriminability yields

$$D_l(\sigma) = \frac{B_l}{B_l + \sigma^2 W_l}.$$

PROOF OF THEOREM 1

Proof. (i) Applying Proposition 2 together with Proposition 4, we obtain

$$\mathrm{TotPers}\big(\Delta_l(\sigma)\big) = \mathcal{L}\big(\Delta_l(\sigma)\big) = \mathcal{L}\big(\{\sigma\xi_l^i\}_{i=1}^N\big) = \mathcal{L}(\sigma\Xi) = \sigma\mathcal{L}(\Xi).$$

Therefore, $\mathrm{TotPers}(\Delta_l(\sigma))$ is strictly increasing since

$$\frac{\partial\ \mathrm{TotPers}(\Delta_l(\sigma))}{\partial\sigma} = \mathcal{L}(\Xi) > 0.$$

(ii) Differentiating the discriminability operator $D_l(\sigma)$ yields

$$\frac{\partial D_l(\sigma)}{\partial\sigma} = \frac{-2B_l\sigma W_l}{\left(B_l + \sigma^2 W_l\right)^2} < 0.$$

where the strict inequality follows from the assumptions $B_l, W_l > 0$ and $\sigma > 0$. Hence, $D_l(\sigma)$ is strictly decreasing. □

Remark 1 (The family under re-standardization)**.** *If instead the family is generated at activation level and equation 2 is re-applied, the coordinate-wise pooled variance becomes* $\left(\mu_{\delta,k}^{\mathrm{raw}}\right)^2/4 + \sigma^2 s_k^2$ *with* $s_k := \bar{W}_{l,k}^{1/2}$ *the pooled within-class standard deviation of Proposition 1, and the standardized displacements differ by*

$$\big(\delta^i(\sigma) - \delta^j(\sigma)\big)_k = \frac{\sigma\left(\xi_k^{\mathrm{raw},i} - \xi_k^{\mathrm{raw},j}\right)}{\sqrt{(\mu_{\delta,k}^{\mathrm{raw}})^2/4 + \sigma^2 s_k^2}},$$

the mean displacement cancelling, where $\xi^{\mathrm{raw},i}$ *is the raw residual displacement. Writing* $m_k := \mu_{\delta,k}^{\mathrm{raw}}/2$*, the coordinate factor* $g_k(\sigma) = \sigma(m_k^2 + \sigma^2 s_k^2)^{-1/2}$ *has*

$$g_k'(\sigma) = \frac{m_k^2}{\left(m_k^2 + \sigma^2 s_k^2\right)^{3/2}} \ \geq\ 0,$$

with equality exactly on the coordinates where $\mu_{\delta,k}^{\mathrm{raw}} = 0$*, on which* $g_k \equiv 1/s_k$*. Every pairwise distance is therefore non-decreasing in* σ*, and since* $\mathcal{L}$ *is non-decreasing in each entry of the distance matrix, so is* $\mathrm{TotPers}$*; it increases strictly as soon as every pair of residual displacements differs on some coordinate with* $\mu_{\delta,k}^{\mathrm{raw}} \neq 0$*, which is the generic case. It is now bounded, with* $\mathrm{TotPers} \to \mathcal{L}(H_l)$ *as* $\sigma \to \infty$*, the separation-free score of Proposition 1, so the linear law* $\mathrm{TotPers}(\Delta(\sigma)) = \sigma\mathcal{L}(\Xi_l)$ *holds only for the family of the theorem.*

For D_l *the situation is different, because it is a ratio of quadratic forms and is therefore unchanged by a common rescaling of all coordinates. Whenever the re-standardization acts as a single scalar,* $m_k^2 + \sigma^2 s_k^2$ *independent of* k*, as in the isotropic model of Appendix O, the factor cancels from numerator and denominator and the reduction* $D_l(\sigma) = B_l/(B_l + \sigma^2 W_l)$ *continues to hold* exactly*, so Theorem 1 applies verbatim. When the rescaling is coordinate-dependent, it does not cancel;* D_l *is no longer that ratio, and we do not prove monotonicity; in simulation it remains strictly decreasing. In either case the orderings remain opposite. This is the same coupling as Proposition 1.*

E FORMAL ANALYSIS OF THE ANSWER-POLARITY AXIS

This appendix expands and provides a formal analysis underlying Sec. 5.3.

Structural decomposition. In the persona format, each contrastive pair shares the same prompt–a statement describing the target behavior together with a question–and differs only in the appended response: the matching answer for x^+ and the non-matching answer for x^-. Motivated by this construction, we model each activation difference as the sum of a context term, a response-token term, and a behavior term:

$$\delta_l^i \ =\ \underbrace{(c_l^i - c_l^i)}_{=0} + \underbrace{\pm t_l}_{\text{answer polarity}} + \underbrace{b_l^i}_{\text{behavior}}. \tag{18}$$

equation 18 is an ansatz suggested by the contrastive construction, rather than a derived identity. All results below are therefore conditional on this decomposition. The sign of t_l records which response token carries the matching answer for example i; it is thus a property of how the dataset is written, rather than of the behavior being steered. We estimate the axis by the half-difference of the answer-conditioned displacement means,

$$t_l := \tfrac{1}{2}\big(\bar{\delta}_l^{\text{yes}} - \bar{\delta}_l^{\text{no}}\big), \qquad \hat{t}_l := t_l/\|t_l\|, \tag{19}$$

Two assumptions enter here. The $\pm t_l$ terms cancel in $\mu_{\delta,l}$ only if the *Yes/No* split is exactly balanced, which the dataset construction of (Sun et al., 2025) enforces; and equation 19 is unbiased for t_l only if the behavior terms b_l^i have equal means across the two answer groups. For a balanced dataset, the mean displacement is therefore

$$\mu_{\delta,l} = \frac{1}{N}\sum_{i=1}^{N} \delta_l^i = \frac{1}{N}\sum_{i=1}^{N} \big(\pm t_l + b_l^i\big) = \mu_{b,l}. \tag{20}$$

Thus, the answer-polarity component does not affect the mean displacement and consequently does not affect mean-displacement (MD) extraction.

We can remove this component using the orthogonal projector

$$P_{\hat{t}_l} = I - \hat{t}_l \hat{t}_l^\top, \tag{21}$$

We then use this projection to study how removing the answer-polarity component affects the two modes considered below: discriminability, through the resulting change in the steering direction and its associated scatter, and connectivity, through the geometric change in the displacement cloud and its minimum spanning tree. The two effects are analyzed separately in the following sections

Discriminability under projection We now analyze how the discriminability score D_l changes when the answer-polarity axis is projected out. Recall that

$$D_l(v) = \frac{B_l(v)}{B_l(v) + W_l(v)}, \qquad B_l(v) = v^\top S_l^b v, \qquad W_l(v) = v^\top S_l^w v. \tag{22}$$

Since B_l and W_l are quadratic forms, D_l is invariant to rescaling of its direction argument:

$$D_l(cv) = D_l(v), \qquad c \neq 0. \tag{23}$$

We therefore evaluate D_l on unit directions without loss of generality.

Let

$$v = \mu_{\delta,l}, \qquad v' = P_{\hat{t}_l} v, \qquad \hat{v} = \frac{v}{\|v\|}, \qquad \hat{v}' = \frac{v'}{\|v'\|}. \tag{24}$$

Lemma 2 (Relation between a vector and its projection)**.** *Let v be a nonzero vector and $v' = P_{\hat{t}_l} v$. Define $\varepsilon := \langle \hat{v}, \hat{t}_l \rangle$ and assume that $|\varepsilon| < 1$, Then,*

$$\hat{v} = \varepsilon \hat{t}_l + \sqrt{1-\varepsilon^2}\, \hat{v}' \tag{25}$$

Proof. $P_{\hat{t}_l}\hat{v} = \hat{v} - \varepsilon \hat{t}_l \neq 0$ and $\|\hat{v} - \varepsilon \hat{t}_l\|^2 = 1 - 2\varepsilon^2 + \varepsilon^2 = 1 - \varepsilon^2$, so

$$\hat{v}' = \frac{\hat{v} - \varepsilon \hat{t}_l}{\sqrt{1-\varepsilon^2}}, \qquad \text{equivalently} \qquad \hat{v} = \varepsilon \hat{t}_l + \sqrt{1-\varepsilon^2}\, \hat{v}'. \tag{26}$$

The pair $(\hat{t}_l, \hat{v}')$ is orthonormal, because $\langle \hat{t}_l, \hat{v}' \rangle = (\langle \hat{t}_l, \hat{v} \rangle - \varepsilon)/\sqrt{1-\varepsilon^2} = 0$. Note also $\langle \hat{v}, \hat{v}' \rangle = \sqrt{1-\varepsilon^2}$. □

The key observation is that projecting the data does not change either the between-class or within-class scatter matrix, S_l^b and S_l^w, when they are evaluated along v'. This gives the following identity.

Lemma 3 (Projection identity)**.** *Let D_l the discriminability and $P_{\hat{t}_l}$ the orthogonal projector onto the subspace orthogonal to $\hat{t}_l$. For any direction v, let $v' = P_{\hat{t}_l} v$. Evaluating discriminability after projecting the activations along v' gives the same value as evaluating the original discriminability along v'. Hence, the shift $D_l(v') - D_l(v)$ is therefore a change of direction, not of data.*

Proof. Consider either the between-class or within-class scatter matrix $S \in \{S_l^b, S_l^w\}$, which are formed from the activations. Applying the projection $P_{\hat{t}_l}$ to the activations, the corresponding scatter matrix becomes $P_{\hat{t}_l} S P_{\hat{t}_l}$. Therefore, evaluating along v' and applying $P_{\hat{t}_l} v' = v'$,

$$(v')^\top (P_{\hat{t}_l} S P_{\hat{t}_l})(v') = (P_{\hat{t}_l} v')^\top S \, (P_{\hat{t}_l} v') = (v')^\top S \, v'.$$

Therefore, the projection does not require redefining the discriminability criterion on a new displacement cloud: its effect can be studied entirely as a change in steering direction from v to v'. □

We now specialize these observations to the steering direction $v = \mu_{\delta,l}$. For notational convenience, and at variance with Appendix D, where the same symbols are evaluated at $\mu_{\delta,l}$ rather than at its unit vector (the two differ by the factor $\|\mu_{\delta,l}\|^2$, which cancels in D_l), let

$$B_l := B_l(\hat{v}), \qquad W_l := W_l(\hat{v}), \qquad B_l' := B_l(\hat{v}'), \qquad W_l' := W_l(\hat{v}'). \tag{27}$$

Similarly, write

$$D_l := D_l(\hat{v}), \qquad D_l' := D_l(\hat{v}'). \tag{28}$$

Proposition 5 (Discriminability under projection)**.** *Let*

$$v = \mu_{\delta,l}, \qquad v' = P_{\hat{t}_l} v,$$

and define

$$\varepsilon := \langle \hat{v}, \hat{t}_l \rangle, \qquad \lambda := 1 - \frac{W_l'}{W_l}.$$

Assume $B_l > 0$, $W_l' > 0$, and $|\varepsilon| < 1$. Then:

(i) $B_l' = (1 - \varepsilon^2)\, B_l$.

(ii) The shift in discriminability is

$$D_l' - D_l = \frac{B_l \, W_l \, (\lambda - \varepsilon^2)}{(B_l + W_l)\big[(1-\varepsilon^2)B_l + (1-\lambda)W_l\big]},$$

and is therefore positive if and only if $\lambda > \varepsilon^2$.

(iii) λ is not free: it is determined by ε and S_l^w through

$$\lambda \, W_l \;=\; \varepsilon^2 \big(W_l(\hat{t}_l) - W_l'\big) \;+\; 2\varepsilon \sqrt{1-\varepsilon^2}\; \hat{t}_l^\top S_l^w \, \hat{v}'.$$

In particular, if $\varepsilon = 0$ then $\hat{v}' = \hat{v}$, $\lambda = 0$, and the shift vanishes identically.

(iv) If $0 < |\varepsilon|$ and $\hat{t}_l$ is an eigenvector of S_l^w, the cross term vanishes and, with $r := W(\hat{t}_l)/W_l'$,

$$\lambda = \frac{\varepsilon^2 (r-1)}{\varepsilon^2 (r-1) + 1}, \qquad \text{and} \qquad D_l' - D_l > 0 \iff r > 1 + \frac{1}{1-\varepsilon^2}.$$

As $\varepsilon \to 0$ at fixed r,

$$D_l' - D_l \;=\; \frac{B_l \, W_l \, (r-2)}{(B_l + W_l)^2} \, \varepsilon^2 \;+\; O(\varepsilon^4).$$

Proof. (i). By Appendix D, $S^b = \frac{N}{2} \mu_{\delta,l} \mu_{\delta,l}^\top = \frac{N}{2} v \, v^\top$, so for a unit vector w

$$B(w) = \tfrac{N}{2} \langle v, w \rangle^2 = \tfrac{N}{2} \|v\|^2 \langle \hat{v}, w \rangle^2.$$

Taking $w = \hat{v}$ gives $B(\hat{v}) = \frac{N}{2}\|v\|^2$, and taking $w = \hat{v}'$ with $\langle \hat{v}, \hat{v}' \rangle = \sqrt{1-\varepsilon^2}$ by Lemma 2 gives $B(\hat{v}') = (1-\varepsilon^2) B_l$. This is an identity, not an approximation.

(ii). We have that

$$D_l' - D_l = \frac{B_l' W_l - B_l W_l'}{(B_l' + W_l')(B_l + W_l)}.$$

Substituting $B_l' = (1-\varepsilon^2)B_l$ from (i) and $W_l' = (1-\lambda)W_l$ from the definition of λ, gives the stated expression. The denominator $(B_l' + W_l')(B_l + W_l)$ is strictly positive under the standing assumptions, so the sign of the shift is the sign of $\lambda - \varepsilon^2$.

(iii). Expanding W_l in the frame equation 26 from Lemma 2,

$$W_l = \varepsilon^2 W(\hat{t}_l) + 2\varepsilon\sqrt{1-\varepsilon^2}\,\hat{t}_l^\top S^w \hat{v}' + (1-\varepsilon^2)W_l'.$$

Subtracting W_l' from both sides and using $\lambda W_l = W_l - W_l'$ yields the identity. If $\varepsilon = 0$ then $\hat{v}' = \hat{v}$ by equation 26, so $\lambda = 0$ and, by (ii), $D_l' = D_l$.

(iv). If $S^w \hat{t}_l = W(\hat{t}_l)\hat{t}_l$ then $\hat{t}_l^\top S^w \hat{v}' = W(\hat{t}_l)\langle \hat{t}_l, \hat{v}'\rangle = 0$, so the expansion in (iii) reduces to

$$W_l = \varepsilon^2 W(\hat{t}_l) + (1-\varepsilon^2)W_l', \qquad \lambda W_l = \varepsilon^2\big(W(\hat{t}_l) - W_l'\big).$$

Dividing both by $W_l' > 0$ and writing $r = W(\hat{t}_l)/W_l'$,

$$\lambda = \frac{\varepsilon^2(r-1)}{\varepsilon^2 r + 1 - \varepsilon^2} = \frac{\varepsilon^2(r-1)}{\varepsilon^2(r-1)+1}.$$

Put $s := \varepsilon^2(r-1)$, so $\lambda = s/(s+1)$ with $s > -1$. By (ii) the shift is positive iff $\lambda > \varepsilon^2$, i.e. iff $s > \varepsilon^2(s+1)$, i.e. iff $s(1-\varepsilon^2) > \varepsilon^2$. Substituting s and dividing by $\varepsilon^2 > 0$ gives $(r-1)(1-\varepsilon^2) > 1$, that is $r > 1 + (1-\varepsilon^2)^{-1}$. Finally, substituting $\lambda = \varepsilon^2(r-1) + O(\varepsilon^4)$ into the expression of (ii) and letting $\varepsilon \to 0$ gives $\lambda - \varepsilon^2 = \varepsilon^2(r-2) + O(\varepsilon^4)$ and denominator $(B_l + W_l)^2 + O(\varepsilon^2)$, hence the stated expansion. □

Connectivity under projection We now analyze how projecting out the answer-polarity axis affects the connectivity of the displacement cloud. In contrast to discriminability, which depends on quadratic forms evaluated along a steering direction, connectivity depends directly on the Euclidean lengths of the edges of a minimum spanning tree (MST) (Appendix C). Consequently, the effect of the projection is geometric: an edge loses precisely the component of its length along the projected-out direction.

In our setting, we apply the projection with $u = \hat{t}_l$ to the displacement cloud Δ_l, which gives the polarity-projected cloud

$$\Delta_l' := P_{\hat{t}_l}\Delta_l = \big\{\delta_l^i - \langle \delta_l^i, \hat{t}_l\rangle\,\hat{t}_l\big\}_{i=1}^N. \tag{29}$$

Proposition 6 (Connectivity under projection)**.** *Let u be a unit vector, $P_u := I - uu^\top$ the orthogonal projector onto $u^\perp$, and $X \subset \mathbb{R}^d$ a point cloud of N points. Then:*

(i) $\mathcal{L}(P_u X) \le \mathcal{L}(X)$.

(ii) Let $T' = \mathrm{MST}(P_u X)$ and lift each of its edges to the corresponding pair of points of X. Then

$$\mathcal{L}(X) - \mathcal{L}(P_u X) \;\le\; \sum_{e \in T'} \frac{\langle e, u\rangle^2}{\|e\|}.$$

In particular, if every lifted edge satisfies $|\langle e, u\rangle| \le \eta$ and $\|e\| \ge w_{\min}$, then $\mathcal{L}(X) - \mathcal{L}(P_u X) \le (N-1)\eta^2/w_{\min}$.

Proof. (i) P_u is 1-Lipschitz and $P_u\,\mathrm{MST}(X)$ is a connected spanning subgraph of $P_u X$ (possibly with coincident vertices), so

$$\mathcal{L}(P_u X) \;\le \sum_{e \in \mathrm{MST}(X)} \|P_u e\| \;\le\; \mathcal{L}(X).$$

(ii) The lift of $T' = \mathrm{MST}(P_u X)$ is a spanning tree of X, hence $\sum_{e\in T'}\|e\| \ge \mathcal{L}(X)$. For each edge, $\|P_u e\| = \sqrt{\|e\|^2 - \langle e, u\rangle^2} \ge \|e\| - \langle e, u\rangle^2/\|e\|$, so

$$\mathcal{L}(P_u X) = \sum_{e\in T'}\|P_u e\| \;\ge\; \sum_{e\in T'}\|e\| - \sum_{e\in T'}\frac{\langle e, u\rangle^2}{\|e\|} \;\ge\; \mathcal{L}(X) - \sum_{e\in T'}\frac{\langle e, u\rangle^2}{\|e\|}.$$

The final clause follows by bounding each of the $N-1$ summands by $\eta^2/w_{\min}$. □



Figure 5: **Sensitivity to the answer-polarity axis**, full version of Figure 4 (Llama-8B, 700 pairs); Alliance Building shown in (a), (b) and (e). (a) D along v^{raw} before and after the axis is projected out, with each curve's top-5 layers marked. (b) The same for $\mathcal{L}$, whose top-5 set is unchanged. (c) The shift in D against the scatter reduction λ of Proposition 5, all six tasks and layers. (d) The largest shift among each criterion's own top-5 layers, divided by its rank-1-to-rank-5 gap. (e) The activations at layer 14 in the plane of the steering direction and the polarity axis: the axis separates the lobes but not the classes. (f) Discriminability along the polarity axis against discriminability along the steering direction, all tasks and depths. The polarity sign of each pair is read off the cloud as the side of the longest spanning-tree edge, which splits every cloud into $350/350$ with the same partition at all 32 layers; the data and script are in `data/polarity/`.

Comparison between two criteria The two propositions capture different effects of removing the answer-polarity axis. For connectivity, the projection acts directly on Euclidean distances and therefore on the MST. In a two-lobe cloud, the resulting change has two contributions: removing the component of the bridge along $\hat{t}_l$, and superimposing the two lobes after projection. The latter changes the MST from spanning two sets of approximately $N/2$ points to spanning N points in the residual space.

Discriminability responds differently because it depends on quadratic forms evaluated along the steering direction. By Proposition 5, the effect of the projection is controlled by $\varepsilon = \langle \hat{\mu}_{\delta,l}, \hat{t}_l \rangle$. It vanishes when the answer-polarity axis is orthogonal to the mean-displacement direction, and becomes positive only when $\hat{t}_l$ carries more than about twice the within-class scatter of the residual steering direction. Thus, connectivity and discriminability need not change by comparable amounts under the same projection: the former is sensitive to the geometry of the entire cloud, whereas the latter is sensitive to how the projected direction interacts with the between- and within-class scatter.

Remark 2 (Effect of Lobe Superposition)**.** *As mentioned above, one of the effects of projecting out $\hat{t}_l$ in the connectivity case is the change from spanning two sets of $N/2$ points to spanning a single set of N points. To quantify this effect, let D_{int} denote the* intrinsic dimension *of a lobe, i.e., the effective number of dimensions in which its points exhibit substantial geometric variation, rather than the ambient dimension of the activation space. By the Beardwood–Halton–Hammersley*

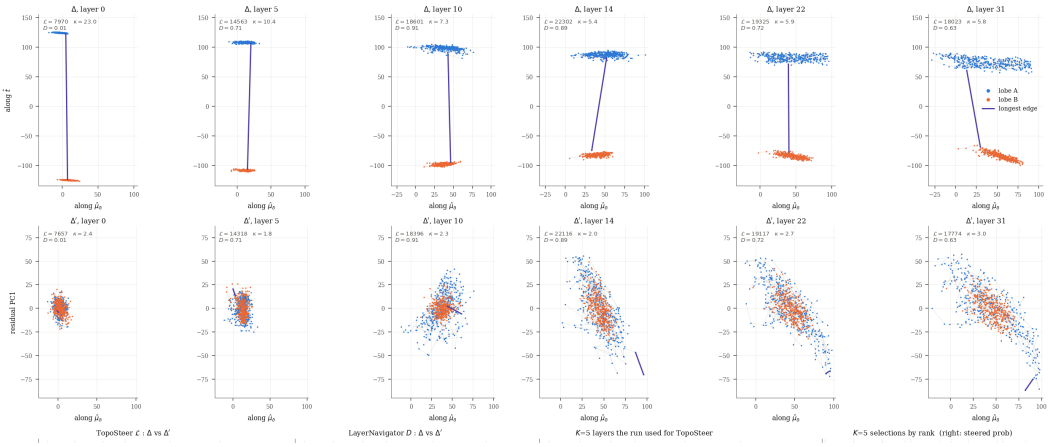

Figure 6: **Anatomy of a displacement cloud** (conscientiousness, Llama-8B, 700 pairs). *Top:* Δ_l at six depths, in the plane spanned by the mean displacement $\hat{\mu}_{\delta,l}$ and the answer-polarity axis $\hat{t}_l$; the cloud splits into two lobes and the longest spanning-tree edge (dark) bridges them. The lobes are point-like at layer 0 and swell with depth, which is where $\mathcal{L}$ grows. *Bottom:* the same clouds after removing $\hat{t}_l$, plotted against the leading residual principal component; the residual behavior geometry is what survives. Panel annotations give $\mathcal{L}$ and LayerNavigator's D.

scaling $\mathcal{L} \asymp n^{1-1/D_{\text{int}}}$, the second part is a relative change of order

$$1 - 2^{-1/D_{\text{int}}} \approx \frac{\log 2}{D_{\text{int}}},$$

Thus, this contribution is small for high-dimensional lobes but can become substantial when the intrinsic dimension is low, and in neither case controlled by the bridge share. Appendix F verifies the scaling.

F Two-Dimensional Illustrations

This appendix gives the constructions behind Figures 7 and 8.

F.1 The decoupling theorem in 2D

Take $\mu_\delta = (4, 0)$ and let Ξ be $N = 40$ points on a noisy unit ring, split into activations via $\tilde{a}_i^{\pm} = \pm\mu_\delta/2 + \sigma(\pm\xi_i/2 + c_i)$, where the shared pair noise c_i inflates within-class scatter but cancels exactly in $\delta_i = \mu_\delta + \sigma\xi_i$. At $\sigma = 0.6$ the classes look cleanly separated and Fisher rewards the layer ($D = 0.98$) while the displacement MST is short ($\mathcal{L} = 4.8$); at $\sigma = 2.4$ the classes visibly overlap, Fisher drops ($D = 0.78$), yet the displacement ring has grown and $\mathcal{L} = 19.2$ — TopoSteer prefers exactly the layer LayerNavigator penalizes (Figure 7, left and middle columns). Sweeping σ traces the two curves of Theorem 1: TotPers is the exact line $\sigma\mathcal{L}(\Xi)$ and $D = B/(B + \sigma^2 W)$ decays, so the induced orderings are reversed at every pair of scales (right column).

F.2 Reading the MST of a displacement cloud

Example 1 (what a bridge in Δ means)**.** *Because the cloud is the set of* per-pair differences*, and Corollary 1 makes $\mathcal{L}(\Delta)$ blind to its mean, a long bridge edge is* never *the class separation $\|\mu_\delta\|$. It signals instead that the pairs split into subpopulations with distinct displacement modes $\mu_\delta^{(1)}, \mu_\delta^{(2)}$: the model implements the behavioral contrast in (at least) two different ways depending on the example, and the bridge length is approximately the gap between the modes. This is bad news for mean-difference steering specifically: v^{MD} averages the lobes and points into the empty region between them, representing neither mode — a bridge-inflated score would select precisely the layers where the single-vector premise fails (Figure 8b). Two flavors must be distinguished by the sizes of*

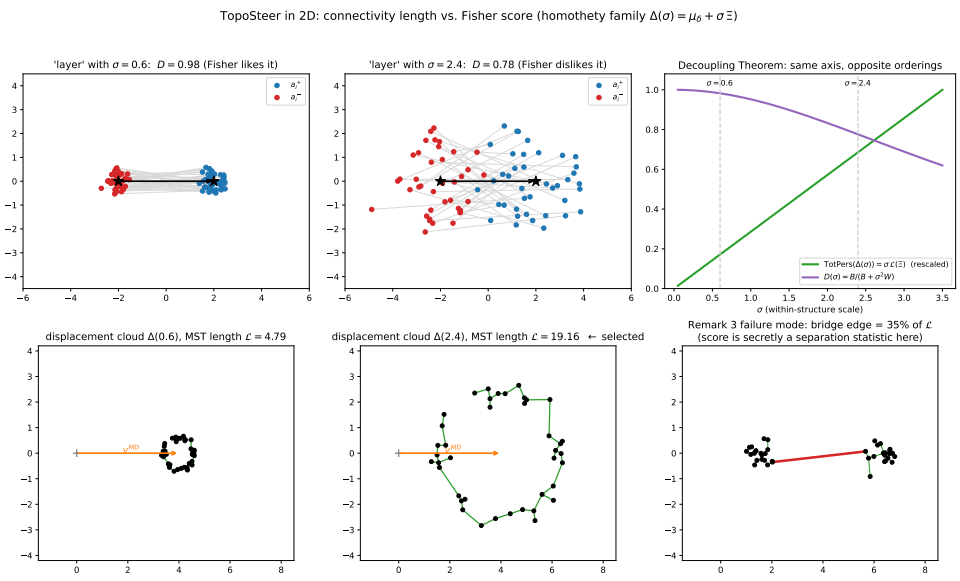

Figure 7: **Separability and displacement connectivity can induce opposite rankings.** The synthetic configurations share the class-mean difference $\mu_\delta = (4, 0)$ and differ only in residual scale σ. Gray segments connect paired positive and negative activations. The displacement clouds keep the same mean direction while their connectivity length grows as $\mathrm{TotPers}(\Delta(\sigma)) = \sigma\mathcal{L}(\Xi)$, whereas discriminability decreases as $D(\sigma) = B/(B + \sigma^2 W)$: the exact ordering reversal of Theorem 1.

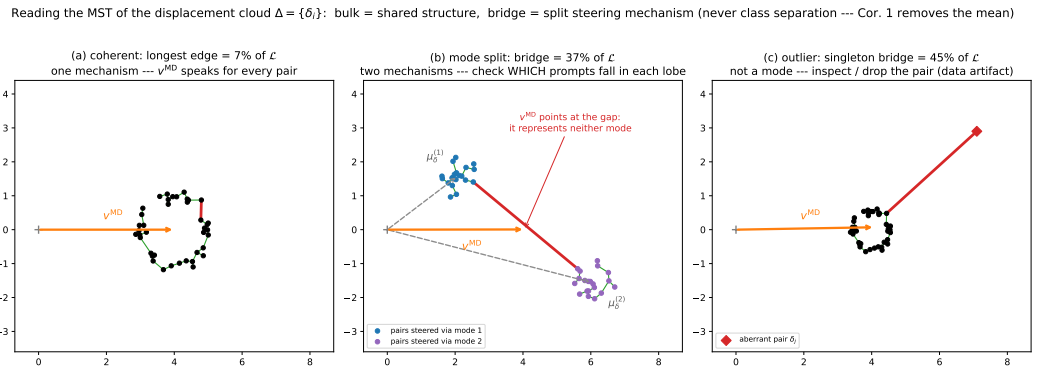

Figure 8: Reading the MST of the displacement cloud Δ (Example 1). (a) Coherent cloud: the longest edge is a small fraction of $\mathcal{L}$ and v^{MD} (orange) represents every pair. (b) Mode split: two subpopulations of pairs (colors) displace along different modes $\mu_\delta^{(1)}, \mu_\delta^{(2)}$ (dashed); the bridge (red) carries a large share of $\mathcal{L}$ and v^{MD} points into the gap between the lobes, representing neither. (c) Outlier: the bridge joins a singleton — a data artifact, not a mode. The bridge share equation 8 separates these regimes and must be reported at score-peak layers.

the components the bridge joins: a bridge between two sizable clusters is a genuine mode split (inspect which prompts fall in each lobe — answer polarity, phrasing templates, and topical subgroups are common culprits), whereas a bridge to a singleton is an outlier pair, i.e. a data or tokenization artifact to inspect or drop (Figure 8c). Finally, Z-scoring rescales coordinates by the total per-coordinate std, and the lobe-gap direction contributes within-class variance, so normalization can dampen or amplify a bridge; the bridge share must therefore be measured in the same normalization used for extraction and is cheap to report in both conventions. In this sense the bridge share is the topological counterpart of LayerNavigator's consistency score, but it is diagnostic rather than scalar: it identifies which *pairs disagree and by how much.*

F.3 THE LOBE-MERGING TERM OF REMARK 2

Remark 2 attributes part of the drop in $\mathcal{L}$ under projection to the two lobes being superimposed rather than to the bridge. We check the predicted size. For each ambient dimension D, we draw two standard Gaussian lobes of $N/2 = 200$ points each, separated along u by a gap large enough that the bridge is a single edge, remove the bridge from both sides of the comparison, and report the residual deficit as a fraction of the bulk length. The prediction is $1 - 2^{-1/D_{\text{int}}}$ with $D_{\text{int}} = D$. Table 4 is produced by `bhh_lobe_merging.py`, provided in the supplement (12 trials per dimension, seed 11).

Table 4: Bulk deficit under projection, against the Beardwood–Halton–Hammersley prediction.

D	$1 - 2^{-1/D}$	measured	s.d.
2	0.293	0.279	0.014
3	0.206	0.193	0.011
5	0.129	0.123	0.008
10	0.067	0.070	0.005
20	0.034	0.042	0.002

The residual gap at $D = 20$ is the finite-N correction: the asymptotic scaling needs $N \gg 2^D$. The term is therefore negligible at the widths of the displacement clouds we score, and the bridge share equation 8 accounts for the rest.

G WHY MEAN-DIFFERENCE EXTRACTION

Why mean-difference extraction. We deliberately restrict attention to Mean-Difference (MD) extraction for three reasons. (i) *Methodological alignment*: our layer-selection criterion is a diagnostic of precisely the premise underlying MD steering — that a single vector, the mean of the displacement cloud, faithfully represents every pair's displacement (Corollary 1, Example 1); probe- and subspace-based extractors rest on different premises and would require different diagnostics. (ii) *Experimental control*: LayerNavigator (Sun et al., 2025) likewise selects layers for MD-extracted vectors, so fixing the extractor isolates the variable under study — layer selection — from extraction effects. (iii) *Standard practice*: MD is the many-pair generalization of activation addition, and the extractor used by contrastive activation addition is hyperparameter-free, and prior evidence indicates difference-in-means directions are causally effective for intervention relative to probe-derived directions (Marks & Tegmark, 2024; Arditi et al., 2024). We make no claim that MD is uniformly the best extractor; under a two-Gaussian model, the Bayes-optimal linear direction is the whitened difference $\Sigma^{-1}\mu_\delta$, and extending the topological diagnostic to whitened, subspace-, and probe-based extractors is left to future work.

H COMPUTATIONAL PROPERTIES

TOPOSTEERintroduces no additional LLMs forward passes beyond those needed to extract the layerwise MD steering vectors. A forward pass for each contrastive input yields activations from all layers simultaneously, and these activations are cached for both vector extraction and layer scoring.

After activation extraction, coordinate-wise normalization and displacement construction require

$$O(LNd) \tag{30}$$

arithmetic operations. A dense pairwise-distance implementation requires

$$O(LN^2d) \tag{31}$$

time to construct all layerwise distance matrices. Using a dense MST routine such as Prim's algorithm adds

$$O(LN^2) \tag{32}$$

time. When layers are processed sequentially, the additional working memory is

$$O(Nd + N^2), \tag{33}$$

excluding the cached model activations. Sorting the L layer scores requires $O(L \log L)$ time.

The dominant cost is therefore an offline geometric computation over the extraction data, rather than additional LLMs inference. Once the ranking has been computed, deployment requires neither persistent-homology computation nor an auxiliary selector model for each new input. The only additional inference-time operation is the vector addition in equation 7, with cost proportional to the number of selected layers and intervention positions.

I Experimental Details

I.1 Setup

Tasks and Datasets. We evaluate global layer selection on six binary behavioral steering tasks from the Anthropic Persona Dataset (Perez et al., 2023): Conscientiousness, Religion Following, Self-Awareness, Self-Improvement, Alliance Building, and Impact Maximization. Each example contains a question paired with two candidate responses, one aligned with the target behavior and one aligned with its opposite. Each behavior contains approximately 1000 examples. We use approximately 700 paired examples to extract the layerwise MD vectors and compute the layer scores, and reserve 100 examples for evaluation.

Models. We evaluate five instruction-tuned LLMs spanning two model families and multiple scales: Llama-3-3B-Instruct, Llama-3-8B-Instruct (Grattafiori et al., 2024), Qwen2.5-3B-Instruct, Qwen2.5-7B-Instruct, and Qwen2.5-32B-Instruct (Team et al., 2024). All interventions are applied at inference time without modifying model parameters.

Baselines. Our primary comparison is LayerNavigator (Sun et al., 2025), the closest prior method for validation-free global layer selection. LayerNavigator scores each layer as $s_\ell = d_\ell + c_\ell$, combining a discriminability term with a consistency term, and steers the top-K layers. The discriminability term is a Fisher-style quotient, $d_\ell = (v_\ell^\top S_b\, v_\ell)/(v_\ell^\top S_t\, v_\ell)$, where v_ℓ is the unit-normalized steering direction and S_b, S_t are the between-class and total scatter matrices of the standardized activations; it is large when the two classes separate along v_ℓ. Since d_ℓ is the term most directly comparable to $\mathcal{L}$—both score a layer by how the contrastive pair geometry is arranged there—we also report **Disc**, which ranks layers by d_ℓ alone.

We include two further references. **Random** draws K distinct layers uniformly without replacement from the middle third of the stack, $[\lfloor L/3 \rfloor, \lfloor 2L/3 \rfloor)$, under a fixed seed; the restricted pool makes this a competitive rather than a degenerate baseline. **Base** is the unsteered model, which bounds how much of any gain is attributable to steering at all. All methods use identical mean-difference steering vectors, the same per-layer strength $\alpha = 1$, and the same budgets $K \in \{1, 2, 3\}$ applied simultaneously in one forward pass; they differ only in which layers they select.

Evaluation Metrics. We measure behavioral steering using the average probability assigned to the response aligned with the target behavior. We additionally report generation perplexity to quantify the effect of steering on fluency. Higher aligned-answer probability and lower perplexity are preferred.

Implementation Details. We extract residual-stream activations from the final prompt token at every model layer. The layerwise MD vectors are computed from the raw activations, while TopoSteerscores are computed from the standardized displacement clouds defined in equation 3. At inference time, we inject the layer-specific vector v_l at each selected layer $l \in S_K$ and at all intervention token positions. Unless otherwise stated, we use steering strength $\alpha = 1.0$ and evaluate $K \in \{1, 2, 3\}$ under both intervention regimes of Sec. 3.3; at $K = 1$ the two regimes coincide, so it is reported once. All filtrations are Euclidean.

Computational Resources. All experiments were conducted using CUDA 12.8 on a single NVIDIA L40S and NVIDIA H100 GPU within a shared CPU Slurm cluster.

Codebase. Our implementation builds on the codebase of LayerNavigator (Sun et al., 2025).

J Detailed Alignment Results

In this section, we provide detailed performance alignment under different variants of scoring metrics.

J.1 Alignment performance under total persistence score

Table 5 reports per-task alignment probability and perplexity for every selector under simultaneous steering; Table 8 reports the sequential regime.

J.2 Sequential Steering with Distinct Layers

Table 8 reports sequential steering results with distinct layers enforced at each selection step.

K Separation-Free Standardization

Proof of Proposition 1. The variance identity is the law of total variance for a balanced two-component mixture whose component means are $\pm\mu^{\mathrm{raw}}_{\delta,l}/2$ about the pooled mean, with $\bar{W}_{l,k}$ and $\sigma^2_{l,k}$ both taken with the same (biased) denominator. Since equation 2 rescales coordinate k by $\sigma^{-1}_{l,k}$ and the displacement is formed afterwards, $\xi^i_{l,k} = \xi^{\mathrm{raw},i}_{l,k}/\sigma_{l,k} = \eta^i_{l,k}(1+\rho_{l,k})^{-1/2}$, and Corollary 1 gives $\mathcal{L}(\Delta_l) = \mathcal{L}(\Xi_l) = \mathcal{L}(H_l\Lambda_l)$. Increasing $\rho_{l,k}$ multiplies coordinate k of every point by a factor in $(0,1)$, hence does not increase any pairwise distance; $\mathcal{L}$ is non-decreasing in each entry of the distance matrix, so $\mathcal{L}(\Delta_l)$ does not increase, strictly so when coordinate k separates at least two points (some spanning-tree edge must then straddle two distinct values of that coordinate). The isotropic case is the scalar $\Lambda_l = (1+\rho_l)^{-1/2} I$ together with Proposition 4. □

Proposition 1 shows that the pooled standardization equation 2 makes $\mathcal{L}(\Delta_l)$ decrease in the coordinate-wise separation-to-noise ratios $\rho_{l,k}$, so a layer can be ranked lower than another purely because its classes are further apart. D_l is invariant to a common scalar rescaling but not to this coordinate-wise one, so both criteria move under the shared standardization; the proposition isolates the effect on $\mathcal{L}$, which is the one we can remove.

The effect can be separated from displacement geometry by dividing each coordinate by the pooled *within-class* standard deviation $\bar{W}^{1/2}_{l,k}$ rather than by $\sigma_{l,k}$, which by Proposition 1 gives the separation-free score $\mathcal{L}(H_l)$; this reuses the cached activations and adds no forward passes. All results in this paper use the pooled standardization equation 2, which is also what LayerNavigator's scoring applies, so that the two criteria are compared on the same representation; comparing the rankings under the two standardizations is left to future work.

L Detail Layer Selection

The layers selected by each strategy on Qwen-3B, for every task and budget K, are given in Table 9.

M Additional Analysis

Table 10 reports detailed MMLU utility results for $K \in \{1, 2\}$, while Table 11 reports the corresponding results for $K = 3$.

Model	K	Method	Conscientiousness		Religion		Self-Improvement		Self-Aware		Alliance		Impact	
			Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)
Llama-3B	$K=0$	Base	72.51	26.82	55.68	21.40	70.96	25.13	59.77	30.25	75.56	26.38	65.89	25.00
	$K=1$	D-only	74.13	26.91	62.90	21.69	74.00	25.94	61.25	30.89	82.16	26.59	69.87	25.22
		Random Selection	73.42	26.80	62.82	21.55	71.51	25.06	60.68	30.30	84.70	26.54	66.63	24.98
		LayerNavigator	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	87.08	26.59	71.01	25.18
		TopoSteer	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	82.17	26.59	69.87	25.22
	$K=2$	D-only	80.18	26.89	81.88	21.63	81.99	25.40	73.92	30.57	86.85	26.67	73.46	25.32
		Random Selection	82.06	26.88	67.61	21.59	82.34	25.44	60.84	30.18	89.56	26.57	70.00	25.32
		LayerNavigator	87.00	26.89	81.88	21.63	81.99	25.40	73.92	30.57	90.00	26.71	73.46	25.32
		TopoSteer	87.00	26.89	81.88	21.63	81.99	25.40	73.92	30.57	90.00	26.71	73.46	25.32
	$K=3$	D-only	84.88	26.90	83.64	21.74	64.51	25.58	74.21	30.69	83.41	26.98	71.51	25.95
		Random Selection	83.20	26.89	82.10	21.64	61.25	26.51	72.29	30.45	71.30	27.59	70.52	26.07
		LayerNavigator	87.88	26.90	85.65	21.64	64.21	25.58	74.60	30.55	70.77	27.08	67.62	25.44
		TopoSteer	86.70	26.90	85.87	21.62	64.21	25.58	74.60	30.55	85.41	26.84	67.62	25.44
Llama-8B	$K=0$	Base	82.06	12.79	68.93	9.46	74.17	12.20	81.86	12.32	81.22	11.08	77.45	9.55
	$K=1$	D-only	83.33	12.79	78.64	9.41	73.05	12.23	82.37	12.30	81.70	11.58	79.30	9.53
		Random Selection	81.89	12.74	71.79	9.37	74.14	12.20	85.42	12.36	89.38	11.06	77.63	9.54
		LayerNavigator	85.34	12.79	78.64	9.41	74.74	12.18	84.50	12.32	81.70	11.08	82.62	9.52
		TopoSteer	87.09	12.78	71.79	9.37	80.11	12.13	88.34	12.40	86.23	11.06	82.62	9.52
	$K=2$	D-only	86.92	12.78	78.98	9.40	74.18	12.18	85.99	12.34	84.30	11.06	79.85	9.53
		Random Selection	85.33	12.79	79.12	9.41	83.23	12.13	85.17	12.36	86.90	11.06	83.24	9.53
		LayerNavigator	90.54	12.73	84.05	9.40	74.18	12.18	91.60	12.43	84.30	11.06	84.54	9.53
		TopoSteer	91.79	12.74	79.23	9.36	88.34	12.10	88.57	12.46	89.50	11.05	85.10	9.52
	$K=3$	D-only	86.36	12.78	79.57	9.38	83.00	12.07	85.81	12.33	84.44	11.06	85.47	9.52
		Random Selection	85.35	12.79	82.65	9.42	84.80	12.15	90.46	12.44	89.02	11.07	82.45	9.53
		LayerNavigator	93.83	12.75	83.61	9.38	83.00	12.07	82.01	12.54	91.76	11.06	84.36	9.53
		TopoSteer	93.83	12.75	77.80	9.37	90.18	12.08	73.29	12.72	88.94	11.27	84.36	9.53
Qwen-3B	$K=0$	Base	69.41	5.06	59.95	4.88	62.58	4.37	57.36	4.59	75.38	5.24	60.51	4.47
	$K=1$	D-only	60.89	5.26	54.73	4.96	59.41	5.17	57.09	4.76	78.36	7.83	56.27	5.42
		Random Selection	69.71	5.00	58.52	4.74	62.76	4.34	55.98	5.53	75.63	5.19	60.67	4.56
		LayerNavigator	61.27	5.34	54.38	5.10	65.13	5.47	64.64	8.60	78.41	7.73	62.00	6.33
		TopoSteer	77.87	5.40	63.51	5.55	72.88	4.89	67.06	5.69	81.91	7.18	67.40	5.17
	$K=2$	D-only	52.05	7.67	50.24	6.41	62.44	6.61	59.01	5.18	77.07	9.24	56.95	7.06
		Random Selection	60.91	5.33	57.90	4.95	62.50	4.41	53.31	5.05	75.35	8.62	61.76	6.07
		LayerNavigator	52.23	7.55	50.31	6.45	62.14	6.63	73.22	11.62	82.03	11.09	65.02	8.13
		TopoSteer	82.35	6.47	66.32	6.15	81.24	5.50	76.00	7.00	85.98	9.14	73.88	8.63
	$K=3$	D-only	51.44	10.06	50.12	7.75	53.34	7.64	52.96	5.91	70.63	9.23	60.00	9.05
		Random Selection	68.96	5.13	55.03	4.63	58.70	5.75	55.68	5.49	74.93	5.23	56.02	5.48
		LayerNavigator	51.58	9.74	50.14	7.86	72.09	8.59	82.96	11.71	80.73	11.47	59.73	8.86
		TopoSteer	84.86	7.61	68.83	6.97	81.49	7.49	81.69	9.49	87.36	10.89	74.02	9.18
Qwen-32B	$K=0$	Base	95.57	3.45	60.77	3.38	82.59	3.76	97.88	3.88	91.97	3.81	67.28	3.69
	$K=1$	D-only	94.72	3.51	63.16	3.41	81.41	3.88	97.97	4.32	92.51	4.65	66.91	3.73
		Random Selection	95.51	3.43	62.44	3.51	82.58	3.76	97.99	4.31	92.96	3.97	67.13	3.70
		LayerNavigator	95.77	4.45	63.16	3.41	83.43	4.11	97.93	4.41	92.51	4.65	67.71	3.99
		TopoSteer	95.74	4.26	60.95	3.40	83.36	4.20	98.19	4.68	93.17	4.51	67.96	4.31
	$K=2$	D-only	95.38	3.84	61.85	3.61	81.43	3.94	97.16	4.57	93.08	5.63	66.14	3.82
		Random Selection	95.39	3.99	61.95	3.58	81.68	3.89	98.05	4.56	92.84	4.65	67.90	4.04
		LayerNavigator	95.90	7.91	61.82	3.61	82.96	5.05	98.04	5.83	92.93	7.57	66.83	3.96
		TopoSteer	95.62	6.31	61.32	3.78	84.14	5.17	97.65	5.95	93.81	6.57	69.60	5.99
	$K=3$	D-only	94.71	4.10	61.01	4.19	80.34	4.28	97.23	5.25	93.27	6.24	64.96	4.01
		Random Selection	94.96	3.94	61.99	4.17	81.59	4.30	97.18	4.50	92.29	4.20	66.64	3.88
		LayerNavigator	95.94	11.03	61.01	4.19	82.97	5.23	97.77	9.72	86.88	10.12	65.90	4.08
		TopoSteer	94.66	10.04	62.23	4.16	83.13	7.68	96.91	9.75	94.72	16.80	70.22	7.54

Table 5: Performance with l2 norm on the difference. Alignment probabilities (Prob, %) and perplexity (PPL, ↓) for different layer-selection strategies under the Euclidean filtration. The leftmost column identifies the target model.

N Detail Performance Evaluation with Evaluation-Based Selection Method

Per-instance baseline (W2S-Multi). We compare TopoSteer against W2S-Multi, the deployable per-instance multi-layer steering method of Nelwan & Wicaksono (2026). The steering setup, prompts, data splits, and metrics are exactly those used for TopoSteer and LayerNavigator: the

Model	K	Method	Conscientiousness		Religion		Self-Improvement		Self-Aware		Alliance		Impact	
			Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)
Llama-3B	$K=0$	Base	72.51	26.82	55.68	21.40	70.96	25.13	59.77	30.25	75.56	26.38	65.89	25.00
	$K=1$	D-only	74.13	26.91	62.90	21.69	74.00	25.94	61.25	30.89	82.16	26.59	69.87	25.22
		Random Selection	73.42	26.80	62.82	21.55	71.51	25.06	60.68	30.30	84.70	26.54	66.63	24.98
		LayerNavigator	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	87.08	26.59	71.01	25.18
		TopoSteer	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	82.17	26.59	69.87	25.22
	$K=2$	D-only	80.18	26.89	81.88	21.63	81.99	25.40	73.92	30.57	86.85	26.67	73.46	25.32
		Random Selection	82.06	26.88	67.61	21.59	82.34	25.44	60.84	30.18	89.56	26.57	70.00	25.32
		LayerNavigator	87.00	26.89	81.88	21.63	81.99	25.40	73.92	30.57	90.00	26.71	73.46	25.32
		TopoSteer	87.00	26.89	81.88	21.63	81.99	25.40	73.92	30.57	90.00	26.71	73.46	25.32
	$K=3$	D-only	84.88	26.90	83.64	21.74	64.51	25.58	74.21	30.69	83.41	26.98	71.51	25.95
		Random Selection	83.20	26.89	82.10	21.64	61.25	26.51	72.29	30.45	71.30	27.59	70.52	26.07
		LayerNavigator	87.88	26.90	85.65	21.64	64.21	25.58	74.60	30.55	70.77	27.08	67.62	25.44
		TopoSteer	86.70	26.90	85.87	21.62	76.52	25.54	74.68	31.10	85.41	26.84	73.78	25.41
Llama-8B	$K=0$	Base	82.06	12.79	68.93	9.46	74.17	12.20	81.86	12.32	81.22	11.08	77.45	9.55
	$K=1$	D-only	83.33	12.79	78.64	9.41	73.05	12.23	82.37	12.30	81.70	11.58	79.30	9.53
		Random Selection	81.89	12.74	71.79	9.37	74.14	12.20	85.42	12.36	89.38	11.06	77.63	9.54
		LayerNavigator	85.34	12.79	78.64	9.41	74.74	12.18	84.50	12.32	81.70	11.08	82.62	9.52
		TopoSteer	87.09	12.78	78.64	9.41	80.11	12.13	89.19	12.39	82.96	11.06	82.62	9.52
	$K=2$	D-only	86.92	12.78	78.98	9.40	74.18	12.18	85.99	12.34	84.30	11.06	79.85	9.53
		Random Selection	85.33	12.79	79.12	9.41	83.23	12.13	85.17	12.36	86.90	11.06	83.24	9.53
		LayerNavigator	90.54	12.73	84.05	9.40	74.18	12.18	91.60	12.43	84.30	11.06	84.54	9.53
		TopoSteer	88.91	12.76	79.23	9.36	80.01	12.11	88.57	12.46	84.30	11.06	84.54	9.53
	$K=3$	D-only	86.36	12.78	79.57	9.38	83.00	12.07	85.81	12.33	84.44	11.06	85.47	9.52
		Random Selection	85.35	12.79	82.65	9.42	84.80	12.15	90.46	12.44	89.02	11.07	82.45	9.53
		LayerNavigator	93.83	12.75	83.61	9.38	83.00	12.07	82.01	12.54	91.76	11.06	84.36	9.53
		TopoSteer	92.20	12.73	82.10	9.37	83.00	12.07	82.01	12.54	91.76	11.06	84.36	9.53
Qwen-3B	$K=0$	Base	69.41	5.06	59.95	4.88	62.58	4.37	57.36	4.59	75.38	5.24	60.51	4.47
	$K=1$	D-only	60.89	5.26	54.73	4.96	59.41	5.17	57.09	4.76	78.36	7.83	56.27	5.42
		Random Selection	69.71	5.00	58.52	4.74	62.76	4.34	55.98	5.53	75.63	5.19	60.67	4.56
		LayerNavigator	61.27	5.34	54.38	5.10	65.13	5.47	64.64	8.60	78.41	7.73	62.00	6.33
		TopoSteer	73.56	6.54	63.51	5.55	59.21	5.32	62.54	5.29	74.42	6.13	55.90	5.50
	$K=2$	D-only	52.05	7.67	50.24	6.41	62.44	6.61	59.01	5.18	77.07	9.24	56.95	7.06
		Random Selection	60.91	5.33	57.90	4.95	62.50	4.41	53.31	5.05	75.35	8.62	61.76	6.07
		LayerNavigator	52.23	7.55	50.31	6.45	62.14	6.63	73.22	11.62	82.03	11.09	65.02	8.13
		TopoSteer	62.26	7.90	66.32	6.15	53.57	6.55	60.43	6.67	65.99	6.64	56.81	6.87
	$K=3$	D-only	51.44	10.06	50.12	7.75	53.34	7.64	52.96	5.91	70.63	9.23	60.00	9.05
		Random Selection	68.96	5.13	55.03	4.63	58.70	5.75	55.68	5.49	74.93	5.23	56.02	5.48
		LayerNavigator	51.58	9.74	50.14	7.86	72.09	8.59	82.96	11.71	80.73	11.47	59.73	8.86
		TopoSteer	51.58	9.74	68.83	6.97	53.33	7.67	72.12	9.19	70.32	9.33	50.27	7.99
Qwen-32B	$K=0$	Base	95.57	3.45	60.77	3.38	82.59	3.76	97.88	3.88	91.97	3.81	67.28	3.69
	$K=1$	D-only	94.72	3.51	63.16	3.41	81.41	3.88	97.97	4.32	92.51	4.65	66.91	3.73
		C-only	95.77	4.45	60.77	3.38	83.08	4.33	97.93	4.41	92.84	4.70	67.71	3.99
		Random Selection	95.51	3.43	62.44	3.51	82.58	3.76	97.99	4.31	92.96	3.97	67.13	3.70
		LayerNavigator	95.77	4.45	63.16	3.41	83.43	4.11	97.93	4.41	92.51	4.65	67.71	3.99
		TopoSteer	95.74	4.26	61.09	3.48	83.36	4.20	98.19	4.33	93.16	4.48	67.91	3.91
	$K=2$	D-only	95.38	3.84	61.85	3.61	81.43	3.94	97.16	4.57	93.08	5.63	66.14	3.82
		C-only	95.56	6.52	61.85	3.61	83.41	5.57	98.01	5.91	93.39	7.34	67.98	4.60
		Random Selection	95.39	3.99	61.95	3.58	81.68	3.89	98.05	4.56	92.84	4.65	67.90	4.04
		LayerNavigator	95.90	7.91	61.82	3.61	82.96	5.05	98.04	5.83	92.93	7.57	66.83	3.96
		TopoSteer	95.62	6.31	61.35	3.57	84.14	5.17	97.97	6.00	93.81	6.57	70.59	5.19
	$K=3$	D-only	94.71	4.10	61.01	4.19	80.34	4.28	97.23	5.25	93.27	6.24	64.96	4.01
		C-only	94.28	10.78	61.01	4.19	83.84	9.35	97.95	12.02	93.84	16.92	68.42	5.80
		Random Selection	94.96	3.94	61.99	4.17	81.59	4.30	97.18	4.50	92.29	4.20	66.64	3.88
		LayerNavigator	95.94	11.03	61.01	4.19	82.97	5.23	97.77	9.72	86.88	10.12	65.90	4.08
		TopoSteer	94.66	10.04	61.62	3.79	83.95	8.63	97.05	9.47	94.04	9.33	70.65	6.34

Table 6: Performance without l2 norm on diff. Alignment probabilities (Prob, %) and perplexity (PPL, ↓) for different layer-selection strategies under the Euclidean filtration. The leftmost column identifies the target model.

contrastive (CAA) steering vectors are extracted from the 700 training pairs of each task, a steering vector is added with α=1 to the last-token residual stream at every selected layer, and alignment (Prob) and perplexity (PPL) are measured on the same 100 test prompts per task. In addition, W2S-

Model	K	Method	Conscientiousness		Religion		Self-Improvement		Self-Aware		Alliance		Impact	
			Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)
Llama-3B	$K=0$	Base	72.51	26.82	55.68	21.40	70.96	25.13	59.77	30.25	75.56	26.38	65.89	25.00
	$K=1$	LayerNavigator	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	87.08	26.59	71.01	25.18
		Ours	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	82.17	26.59	69.87	25.22
	$K=2$	LayerNavigator	87.00	26.89	81.88	21.63	81.99	25.40	73.92	30.57	90.00	26.71	73.46	25.32
		Ours	87.00	26.89	81.88	21.63	81.99	25.40	73.92	30.57	90.00	26.71	73.46	25.32
	$K=3$	LayerNavigator	87.88	26.90	85.65	21.64	64.21	25.58	74.60	30.55	70.77	27.08	67.62	25.44
		Ours	86.70	26.90	85.87	21.62	76.52	25.53	74.68	31.10	85.41	26.84	73.78	25.41
Llama-8B	$K=0$	Base	82.06	12.79	68.93	9.46	74.17	12.20	81.86	12.32	81.22	11.08	77.45	9.55
	$K=1$	LayerNavigator	85.34	12.79	78.64	9.41	74.74	12.18	84.50	12.32	81.70	11.08	82.62	9.52
		TopoSteer	87.09	12.78	78.64	9.41	80.11	12.13	89.19	12.39	82.96	11.06	82.62	9.52
	$K=2$	LayerNavigator	90.54	12.73	84.05	9.40	74.18	12.18	91.60	12.43	84.30	11.06	84.54	9.53
		Ours	88.91	12.76	79.23	9.36	80.01	12.11	88.57	12.46	84.30	11.06	84.54	9.53
	$K=3$	LayerNavigator	93.83	12.75	83.61	9.38	83.00	12.07	82.01	12.54	91.76	11.06	84.36	9.53
		Ours	92.20	12.73	82.10	9.37	83.00	12.07	82.01	12.54	91.76	11.06	84.36	9.53
Qwen-3B	$K=0$	Base	69.41	5.06	59.95	4.88	62.58	4.37	57.36	4.59	75.38	5.24	60.51	4.47
	$K=1$	LayerNavigator	61.27	5.34	54.38	5.10	65.13	5.47	64.64	8.60	78.41	7.73	62.00	6.33
		Ours	73.56	6.54	63.51	5.55	59.21	5.32	62.54	5.29	74.42	6.13	55.90	5.50
	$K=2$	LayerNavigator	52.23	7.55	50.31	6.45	62.14	6.63	73.22	11.62	82.03	11.09	65.02	8.13
		Ours	62.26	7.90	66.32	6.15	53.57	6.55	60.43	6.67	65.99	6.64	56.81	6.87
	$K=3$	LayerNavigator	51.58	9.74	50.14	7.86	72.09	8.59	82.96	11.71	80.73	11.47	59.73	8.86
		Ours	51.58	9.74	68.83	6.97	53.33	7.67	72.12	9.19	70.32	9.33	50.27	7.99
Qwen-32B	$K=0$	Base	95.57	3.45	60.77	3.38	82.59	3.76	97.88	3.88	91.97	3.81	67.28	3.69
	$K=1$	LayerNavigator	95.77	4.45	63.16	3.41	83.43	4.11	97.93	4.41	92.51	4.65	67.71	3.99
		Ours	95.74	4.26	61.09	3.48	83.36	4.20	98.19	4.33	93.16	4.48	67.91	3.91
	$K=2$	LayerNavigator	95.90	7.91	61.82	3.61	82.96	5.05	98.04	5.83	92.93	7.57	66.83	3.96
		Ours	95.62	6.31	61.35	3.57	84.14	5.17	97.97	6.00	93.81	6.57	70.59	5.19
	$K=3$	LayerNavigator	95.94	11.03	61.01	4.19	82.97	5.23	97.77	9.72	86.88	10.12	65.90	4.08
		Ours	94.66	10.04	61.62	3.79	83.95	8.63	97.05	9.47	94.04	9.33	70.65	6.34

Table 7: Performance without l2 norm of difference. Alignment probabilities (Prob, %) and perplexity (PPL, ↓) for different layer-selection strategies under Euclidean distance metrics. The leftmost column identifies the target model. Green values indicate the best performance within each K block, and the small annotation shows the absolute improvement over the competing method.

Multi chooses layers per input rather than per task. On the 200 validation prompts of each task, we first compute the single-layer effect of every layer: the change in the restricted Yes/No probability of the gold answer when that layer alone is steered. This costs $(L{+}1)$ forward passes per prompt (one unsteered pass plus one per layer) and requires the gold labels. Each prompt is then embedded with a sentence encoder (`bge-base-en-v1.5` (Xiao et al., 2024)) and reduced to 25 dimensions by PCA fitted on the validation split. A one-hidden-layer MLP ($25{\to}64{\to}L$, ReLU, dropout 0.3) is trained to reproduce the distribution of single-layer effects over layers, using a listwise KL loss with the effects expressed in percentage points, optimized with full-batch Adam (learning rate 10^{-3}, weight decay 10^{-4}, 300 epochs). At test time, each input is steered at the K layers ranked highest by the MLP, so different inputs can receive different layer sets; we report $K \in \{1, 2, 3\}$ to match TOPOSTEER.

Comparison with Evaluation-Based Selection. Table 2 reports the detailed comparison with the per-instance W2S-Multi baseline. W2S-Multi achieves higher alignment probability on average (81.25% vs. 78.90%), but requires substantially more computation, with a total runtime of 15.9 minutes compared with 8.4 minutes for TOPOSTEER. This difference reflects the additional inference and label-dependent training required by W2S-Multi, whereas TOPOSTEER computes layer scores directly from the extraction data without additional steering evaluations. The full model-level results show the same trade-off across model families, with W2S-Multi generally achieving higher alignment at a higher perplexity and computational cost.

Model	K	Method	Conscientiousness		Religion		Self-Improvement		Self-Aware		Alliance		Impact	
			Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)	Prob (↑)	PPL (↓)
Llama-3B	$K=0$	Base	72.51	26.82	55.68	21.40	70.96	25.13	59.77	30.25	75.56	26.38	65.89	25.00
	$K=2$	D-only	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	82.17	26.59	69.87	25.22
		C-only	82.78	26.89	64.90	21.56	79.98	25.26	66.61	30.29	87.08	26.59	71.01	25.18
		Random Selection	78.65	26.89	60.62	21.54	80.80	25.38	61.78	30.32	89.55	26.59	67.79	25.23
		LayerNavigator	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	87.08	26.59	71.01	25.18
		TopoSteer	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	82.17	26.59	69.87	25.22
	$K=3$	D-only	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	82.17	26.59	69.87	25.22
		C-only	82.78	26.89	64.90	21.56	79.98	25.26	66.61	30.29	87.08	26.59	71.01	25.18
		Random Selection	80.29	26.86	66.14	21.64	84.82	25.44	62.61	30.30	90.37	26.63	71.85	25.29
		LayerNavigator	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	87.08	26.59	71.01	25.18
		TopoSteer	77.16	26.89	64.90	21.56	76.00	25.34	66.61	30.29	82.17	26.59	69.87	25.22
Llama-8B	$K=0$	Base	82.06	12.79	68.93	9.46	74.17	12.20	81.86	12.32	81.22	11.08	77.45	9.55
	$K=2$	D-only	83.33	12.79	78.64	9.41	73.05	12.23	82.37	12.30	81.70	11.08	79.30	9.53
		C-only	87.42	12.78	76.26	9.42	79.38	12.14	89.19	12.39	88.49	11.06	80.66	9.53
		Random Selection	88.21	12.78	71.44	9.42	79.04	12.12	85.23	12.36	87.33	11.06	77.70	9.54
		LayerNavigator	85.34	12.79	78.64	9.41	74.74	12.18	84.50	12.32	81.70	11.08	82.62	9.52
		TopoSteer	87.09	12.78	71.79	9.37	80.11	12.13	88.34	12.40	86.23	11.06	82.62	9.52
	$K=3$	D-only	83.33	12.79	78.64	9.41	73.05	12.23	82.37	12.30	81.70	11.08	79.30	9.53
		C-only	87.42	12.78	76.26	9.42	79.38	12.14	89.19	12.39	88.49	11.06	80.66	9.53
		Random Selection	87.99	12.78	74.77	9.42	81.42	12.10	85.49	12.35	87.25	11.06	81.67	9.53
		LayerNavigator	85.34	12.79	78.64	9.41	74.74	12.18	84.50	12.32	81.70	11.08	82.62	9.52
		TopoSteer	87.09	12.78	71.79	9.37	80.11	12.13	88.34	12.40	86.23	11.06	82.62	9.52
Qwen-3B	$K=0$	Base	69.41	5.06	59.95	4.88	62.58	4.37	57.36	4.59	75.38	5.24	60.51	4.47
	$K=2$	D-only	61.32	5.36	54.30	5.19	59.30	5.19	56.70	4.75	78.31	7.63	55.94	5.49
		C-only	84.46	6.86	59.43	6.14	70.07	4.61	64.69	8.62	80.86	6.99	65.89	7.26
		Random Selection	71.29	6.81	53.38	6.17	61.42	4.59	58.48	4.86	76.10	4.92	60.37	4.43
		LayerNavigator	61.32	5.36	54.30	5.19	69.67	7.91	64.69	8.62	80.83	11.18	61.87	6.21
		TopoSteer	77.84	5.37	63.56	5.55	72.86	4.88	67.06	5.65	81.91	7.18	67.37	5.14
	$K=3$	D-only	61.32	5.36	54.30	5.19	59.30	5.19	56.70	4.75	78.31	7.63	55.94	5.49
		C-only	84.46	6.86	59.43	6.14	70.07	4.61	64.69	8.62	80.86	6.99	65.89	7.26
		Random Selection	71.43	6.91	53.29	6.27	61.17	4.64	58.37	4.94	76.20	4.94	60.02	4.70
		LayerNavigator	61.32	5.36	54.30	5.19	69.67	7.91	64.69	8.62	83.81	11.42	61.87	6.21
		TopoSteer	77.84	5.37	63.56	5.55	72.86	4.88	67.06	5.65	81.91	7.18	67.37	5.14
Qwen-7B	$K=0$	Base	83.50	15.30	84.39	9.80	83.28	10.94	94.21	11.82	87.43	10.61	73.38	9.39
	$K=2$	D-only	86.44	15.92	82.91	8.16	83.97	15.03	95.17	13.90	91.26	23.82	75.85	10.13
		C-only	87.51	16.68	86.29	13.22	85.61	24.65	94.72	14.92	91.72	24.14	73.91	10.93
		Random Selection	85.77	16.62	85.40	8.17	84.30	10.31	95.15	12.95	89.71	12.82	74.70	9.11
		LayerNavigator	84.52	17.42	83.23	11.16	83.94	20.11	94.75	16.52	91.26	23.82	75.85	10.13
		TopoSteer	87.25	17.21	84.81	10.41	84.60	22.62	94.59	14.66	89.63	22.72	74.44	15.09
	$K=3$	D-only	86.44	15.92	82.91	8.16	83.97	15.03	95.17	13.90	91.26	23.82	75.85	10.13
		C-only	87.51	16.68	86.29	13.22	85.61	24.65	94.72	14.92	91.72	24.14	73.91	10.93
		Random Selection	85.72	16.51	87.42	12.55	84.75	14.55	95.18	12.95	90.80	15.74	77.07	9.58
		LayerNavigator	84.52	17.42	83.23	11.16	83.94	20.11	94.75	16.52	91.26	23.82	75.85	10.13
		TopoSteer	87.25	17.21	84.81	10.41	84.60	22.62	94.59	14.66	89.63	22.72	74.44	15.09
Qwen-32B	$K=0$	Base	95.57	3.45	60.77	3.38	82.59	3.76	97.88	3.88	91.97	3.81	67.28	3.69
	$K=2$	D-only	94.73	3.55	62.19	3.60	80.78	3.92	97.99	4.49	92.56	4.59	66.54	3.84
		C-only	95.49	6.61	63.25	3.40	83.08	5.47	97.78	6.84	93.22	6.80	67.89	4.42
		Random Selection	92.94	4.00	65.19	3.84	81.09	3.91	98.08	4.05	92.76	4.36	67.00	3.74
		LayerNavigator	95.49	6.61	62.19	3.60	83.43	4.09	97.96	4.42	92.56	4.59	67.80	4.00
		TopoSteer	95.79	4.35	60.98	3.45	83.35	4.17	97.56	5.64	93.13	4.53	68.70	4.89
	$K=3$	D-only	94.73	3.55	62.19	3.60	80.78	3.92	97.99	4.49	92.56	4.59	65.61	3.91
		C-only	95.49	6.61	63.25	3.40	83.08	5.47	97.78	6.84	93.22	6.80	67.89	4.42
		Random Selection	92.99	3.95	64.66	3.66	80.84	3.90	97.85	4.14	92.56	4.46	67.53	3.73
		LayerNavigator	95.49	6.61	61.87	3.99	83.43	4.09	97.96	4.42	92.56	4.59	67.80	4.00
		TopoSteer	95.79	4.35	60.98	3.45	83.35	4.17	97.13	7.70	93.13	4.53	68.70	4.89

Table 8: Alignment probabilities (Prob, %) and perplexity (PPL, ↓) for different layer-selection strategies under the Euclidean filtration. The leftmost column identifies the target model.

K	Method	Conscientiousness	Religion	Self-Improvement	Self-Aware	Alliance	Impact
1	D-only	21	22	22	20	23	22
	C-only	26	23	34	25	34	25
	Random Selection	12	17	12	23	18	12
	LayerNavigator	21	22	23	25	23	23
	TOPOSTEER (diff)	23	35	22	30	22	22
	TOPOSTEER (ℓ_2)	35	35	35	35	35	35
2	D-only	21, 22	22, 21	22, 23	20, 19	23, 22	22, 23
	C-only	26, 21	23, 24	34, 33	25, 34	34, 35	25, 23
	Random Selection	13, 21	13, 16	15, 18	12, 21	21, 23	17, 23
	LayerNavigator	21, 22	22, 21	23, 22	25, 24	23, 24	23, 24
	TOPOSTEER (diff)	23, 22	35, 34	22, 21	30, 23	22, 21	22, 23
	TOPOSTEER (ℓ_2)	35, 34	35, 34	35, 34	35, 34	35, 34	35, 26
3	D-only	21, 22, 23	22, 21, 23	22, 23, 21	20, 19, 21	23, 22, 21	22, 23, 24
	C-only	26, 21, 25	23, 24, 25	34, 33, 35	25, 34, 33	34, 35, 33	25, 23, 26
	Random Selection	12, 13, 20	14, 17, 21	16, 18, 22	15, 17, 23	12, 14, 17	13, 15, 22
	LayerNavigator	21, 22, 23	22, 21, 23	23, 22, 24	25, 24, 26	23, 24, 22	23, 24, 22
	TOPOSTEER (diff)	23, 22, 21	35, 34, 33	22, 21, 23	30, 23, 35	22, 21, 23	22, 23, 21
	TOPOSTEER (ℓ_2)	35, 34, 33	35, 34, 33	35, 34, 30	35, 34, 33	35, 34, 33	35, 26, 30

Table 9: Layers selected by each strategy on Qwen-3B-Instruct (36 layers, indexed 0–35), for every persona task and layer budget K. Layers are listed in selection order, highest-scoring first; for Random Selection they are the sampled set. TOPOSTEER is shown under both scoring clouds: the standardized difference cloud (diff) and its unit-sphere projection (ℓ_2). All baselines concentrate in the middle of the stack, whereas TOPOSTEER (ℓ_2) consistently selects the final layers.

		$K=1$			$K=2$		
Model	Base	LN	Ours	Ours$_{\ell_2}$	LN	Ours	Ours$_{\ell_2}$
Llama-3B	59.56	60.00 (+0.44)	60.00 (+0.44)	60.00 (+0.44)	58.86 (−0.70)	58.86 (−0.70)	58.86 (−0.70)
Llama-8B	67.89	67.46 (−0.43)	67.37 (−0.52)	67.37 (−0.52)	66.75 (−1.14)	67.81 (−0.08)	66.23 (−1.66)
Qwen-3B	67.19	67.21 (+0.02)	66.93 (−0.26)	67.28 (+0.09)	66.84 (−0.35)	66.75 (−0.44)	67.54 (+0.35)
Qwen-7B	74.39	73.95 (−0.44)	74.21 (−0.18)	74.39 (+0.00)	73.68 (−0.71)	74.56 (+0.17)	74.30 (−0.09)
Qwen-32B	83.86	83.60 (−0.26)	83.95 (+0.09)	83.95 (+0.09)	83.77 (−0.09)	83.60 (−0.26)	83.60 (−0.26)

Table 10: MMLU accuracy (%, ↑) under simultaneous persona steering for $K \in \{1, 2\}$. Each entry reports 5-shot MMLU accuracy, with the absolute change from the unsteered Base model in parentheses (percentage points). Models are steered on `conscientiousness` at $\alpha = 1.0$.

O COMPARING CONNECTIVITY LENGTH VS. SUM OF DISTANCE TO THE MEAN

A natural way to ask "how good is the mean as a representative of the individual points" is to measure the typical distance from a point to some reference — either the mean itself, or, as a proxy for overall compactness, the connectivity length $\mathcal{L}$ captured by the minimum spanning tree. Intuitively, a single mean should represent a unimodal cloud reasonably well (points are dispersed around one center) but should represent a bimodal cloud poorly (no single point, including the mean, sits close to both modes). A good quality measure should therefore be *sensitive to modality*, scoring the unimodal case as better represented and the bimodal case as worse represented.

The relevant standardization here is the one TOPOSTEER actually uses: the pooled $2N$ activations $\{a_i^+\} \cup \{a_i^-\}$ are standardized once, coordinate-wise, using a single mean and std computed over all $2N$ points (Sec. 3.1, Eq. equation 2; Proposition 1 records what that standardization couples the score to; this is also exactly what LayerNavigator's own scoring code performs on `all_acts` before differencing). The displacement cloud $\Delta = \{\delta_i\}$ is formed only afterward, $\delta_i = \tilde{a}_i^+ - \tilde{a}_i^-$, with no further rescaling. This standardization scale is therefore driven by class separation and any pair-shared noise that cancels in the difference but not in the individual activations — quantities the displacement cloud itself cannot see, and this shapes which of the two candidate statistics below remain reliable.

		$K = 3$		
Model	Base	LN	Ours	Ours$_{\ell_2}$
Llama-3B	59.56	56.84 (-2.72)	56.49 (-3.07)	56.49 (-3.07)
Llama-8B	67.89	66.14 (-1.75)	66.58 (-1.31)	66.14 (-1.75)
Qwen-3B	67.19	67.02 (-0.17)	67.02 (-0.17)	67.19 ($+0.00$)
Qwen-7B	74.39	73.60 (-0.79)	74.47 ($+0.08$)	74.39 ($+0.00$)
Qwen-32B	83.86	83.25 (-0.61)	83.77 (-0.09)	83.77 (-0.09)

Table 11: MMLU accuracy (%, ↑) under simultaneous persona steering for $K = 3$. Parenthetical values denote the absolute change from the unsteered Base model in percentage points. Models are steered on `conscientiousness` at $\alpha = 1.0$.

O.1 Model

We extend the homothety family of Sec. 5.2 (Appendix D) to allow the residual shape to be unimodal or bimodal, dropping the layer subscript l as in that example:

$$a_i^+ = \tfrac{\mu_\delta}{2} + \sigma\left(\tfrac{\xi_i}{2} + c_i\right), \qquad a_i^- = -\tfrac{\mu_\delta}{2} + \sigma\left(-\tfrac{\xi_i}{2} + c_i\right),$$

so that $\delta_i = a_i^+ - a_i^- = \mu_\delta + \sigma\xi_i$ exactly, with the pair-shared noise c_i canceling as before. Isotropically:

- $\mu_\delta = (\mu_0/\sqrt{D})\mathbf{1}$, so $\|\mu_\delta\| = \mu_0$ (class separation, spread evenly across coordinates);
- c_i i.i.d. uniform on the sphere of radius γ (shared pair-level noise);
- ξ_i **unimodal**: i.i.d. uniform on the sphere of radius R_u; or **bimodal**: $N/2$ points uniform on the sphere of radius R_w centered at $c_\xi = (\mu_\xi/\sqrt{D})\mathbf{1}$, and $N/2$ centered at $-c_\xi$, the mode-separation spread evenly across coordinates.

Write $V_\xi := R_u^2$ (unimodal) or $V_\xi := R_w^2 + \mu_\xi^2$ (bimodal, by the usual law-of-total-variance decomposition).

O.2 Lemma: sum of norms vs. connectivity length

Lemma 4 (Sum of norms vs. connectivity length under pooled-activation standardization)**.** *Under the model above, as $D \to \infty$ with N fixed, let Δ denote the displacement cloud obtained by pooled standardization of the $2N$ activations followed by differencing, and let*

$$\tau := \sigma^2\left(\frac{V_\xi}{4} + \gamma^2\right) + \frac{\mu_0^2}{4}.$$

Part A (sum of norms). *For either shape,*

$$\mathrm{SN}(\Delta) := \sum_i \|\delta_i - \bar{\delta}\| = N\sigma\sqrt{V_\xi}\,\sqrt{D/\tau} + o(\sqrt{D}),$$

so

$$\frac{\mathrm{SN}(\Delta_{\mathrm{bi}})}{\mathrm{SN}(\Delta_{\mathrm{uni}})} \;\to\; \sqrt{\frac{V_{\xi,\mathrm{bi}}}{V_{\xi,\mathrm{uni}}}} \cdot \sqrt{\frac{\tau_{\mathrm{uni}}}{\tau_{\mathrm{bi}}}}.$$

Part B (connectivity length).

$$\mathcal{L}(\Delta_{\mathrm{uni}}) = (N-1)\sigma\sqrt{2}\,R_u\sqrt{D/\tau_{\mathrm{uni}}} + o(\sqrt{D}),$$

$$\mathcal{L}(\Delta_{\mathrm{bi}}) = \left[(N-2)\sigma\sqrt{2}\,R_w + \sigma\sqrt{4\mu_\xi^2 + 2R_w^2}\right]\sqrt{D/\tau_{\mathrm{bi}}} + o(\sqrt{D}),$$

so

$$\frac{\mathcal{L}(\Delta_{\mathrm{bi}})}{\mathcal{L}(\Delta_{\mathrm{uni}})} \;\to\; \underbrace{\frac{(N-2)R_w + \sqrt{2\mu_\xi^2 + R_w^2}}{(N-1)R_u}}_{\textit{raw shape factor}} \times \underbrace{\sqrt{\tau_{\mathrm{uni}}/\tau_{\mathrm{bi}}}}_{\textit{standardization dilution}}.$$

Part C (bridge share). *Algorithm 1 reports, alongside $\mathcal{L}$, the bridge share $b_l := \max_{e\in \mathrm{MST}(\Delta)} w(e)/\mathcal{L}(\Delta)$ (Eq. equation 8). Being a ratio of a single edge to the total, b_l is invariant to any positive scalar multiplying the whole cloud, hence to τ entirely:*

$$b(\Delta_{\mathrm{uni}}) \to \frac{1}{N-1}, \qquad b(\Delta_{\mathrm{bi}}) \to \frac{\sqrt{2\mu_\xi^2 + R_w^2}}{(N-2)R_w + \sqrt{2\mu_\xi^2 + R_w^2}},$$

regardless of μ_0, γ, σ, with $b(\Delta_{\mathrm{bi}}) > b(\Delta_{\mathrm{uni}})$ whenever $\mu_\xi > 0$.

O.3 Proof of Lemma 4

Step 0 (translation invariance, exact). By Proposition 4, $\mathcal{L}(\Delta) = \mathcal{L}(\Delta - v^{\mathrm{MD}})$ for any D, N, with no asymptotics needed. Centering Δ at its own empirical mean for Part A removes the same constant $v^{\mathrm{MD}} = \mu_\delta$ in the standardized scale. So for both statistics, the effect of μ_δ is confined entirely to how it inflates the standardization scale τ below — never to a residual shift of the centered cloud.

Step 1 (raw, unstandardized distances concentrate to constants). For ξ_i, ξ_j i.i.d. uniform on a sphere of radius R in $\mathbb{R}^D$,

$$\|\xi_i - \xi_j\|^2 = 2R^2 - 2\langle \xi_i, \xi_j \rangle.$$

Writing $\xi_i = R\,u_i$ for u_i uniform on the unit sphere, each coordinate of u_i behaves for large D like an independent $N(0, 1/D)$ variable, so $\langle u_i, u_j\rangle = \sum_k u_{i,k}u_{j,k}$ is a sum of D independent mean-zero terms of variance $O(1/D^2)$ each, giving $\langle u_i, u_j\rangle = O_p(1/\sqrt{D}) \to 0$. Hence $\|\xi_i - \xi_j\| \xrightarrow{p} \sqrt{2}\,R$ — and this holds for *every* pair (i, j), not just in expectation, so in the limit *all* pairwise distances within a radius-R spherical cloud become equal to the same constant. The same computation, applied to $\xi_i = c_\xi + v_i$, $\xi_j = -c_\xi + v_j$ in different modes of the bimodal cloud, gives $\|\xi_i - \xi_j\| \xrightarrow{p} \sqrt{4\mu_\xi^2 + 2R_w^2}$; applied within one mode it gives $\sqrt{2}\,R_w$, as in the unimodal case.

Step 2 (the pooled standardization is a single shared scalar). The pooled set $\{a_i^+\} \cup \{a_i^-\}$ is, per coordinate k, a 50/50 mixture of the class-$+$ and class-$-$ distributions. By the law of total variance,

$$s_k^2 = \underbrace{\tfrac{1}{2}\mathrm{Var}(a_{i,k}^+ \mid +) + \tfrac{1}{2}\mathrm{Var}(a_{i,k}^- \mid -)}_{\text{within-class}} + \underbrace{\mathrm{Var}\big(\{\mu_{\delta,k}/2, -\mu_{\delta,k}/2\}\big)}_{\text{between-class}}.$$

From $a_i^+ = \mu_\delta/2 + \sigma(\xi_i/2 + c_i)$ and independence of ξ_i, c_i: within-class variance is $\sigma^2\big(\mathrm{Var}(\xi_k)/4 + \mathrm{Var}(c_k)\big)$ (identical for class $-$ by the symmetric construction). The between-class term is the variance of a two-point $\pm a$ distribution, which is a^2; here $a = \mu_{\delta,k}/2$, giving $\mu_{\delta,k}^2/4$. So

$$s_k^2 = \sigma^2\left(\mathrm{Var}(\xi_k) + 4\,\mathrm{Var}(c_k)\right)/4 + \mu_{\delta,k}^2/4.$$

Substituting the isotropic per-coordinate values $\mathrm{Var}(\xi_k) = V_\xi/D$, $\mathrm{Var}(c_k) = \gamma^2/D$, $\mu_{\delta,k}^2 = \mu_0^2/D$ gives $s_k^2 = \tau/D$ for *every* coordinate k — the same number regardless of k, because every quantity entering it (V_ξ, γ, μ_0) was assumed spread evenly across coordinates. A coordinate-wise standardization that divides every coordinate by the same number is, by definition, a single global scalar division: $\delta_i - \bar{\delta} = (\mu_\delta + \sigma\xi_i)/s - \mu_\delta/s = \sigma\xi_i/s = \sigma\xi_i\sqrt{D/\tau}$.

Step 3 (Part A). For the unimodal cloud, $\|\xi_i\| = R_u$ exactly for every i (points lie exactly on the sphere), so $\|\delta_i - \bar{\delta}\| = \sigma R_u\sqrt{D/\tau_{\mathrm{uni}}}$ exactly, and summing over N points gives $N\sigma R_u\sqrt{D/\tau_{\mathrm{uni}}}$ — matching the claimed formula since $R_u = \sqrt{V_\xi}$ in the unimodal case. For the bimodal cloud, $\|\xi_i\|^2 = \mu_\xi^2 + R_w^2 \pm 2\langle c_\xi, v_i\rangle$ with the cross term $O_p(\mu_\xi R_w/\sqrt{D}) \to 0$ by the Step 1 argument, so $\|\xi_i\| \to \sqrt{\mu_\xi^2 + R_w^2} = \sqrt{V_\xi}$ for both modes; the $O_p(1)$ fluctuations have mean zero across the two equally sized modes and cancel in the sum, giving $N\sigma\sqrt{V_\xi}\sqrt{D/\tau_{\mathrm{bi}}} + o(\sqrt{D})$.

Step 4 (Part B). Step 1 shows every pairwise distance within the unimodal cloud converges to the *same* constant $\sqrt{2}\,\sigma R_u\sqrt{D/\tau_{\mathrm{uni}}}$ (after applying the Step 2 scalar); when all edges of a complete graph have equal weight, every spanning tree — in particular the minimum one — has the same total weight, namely $(N-1)$ times that constant. This gives the stated formula for $\mathcal{L}(\Delta_{\mathrm{uni}})$. For

the bimodal cloud, the same argument applies separately within each mode (giving $(N/2-1)$ edges of length $\sqrt{2}\,\sigma R_w\sqrt{D/\tau_{\mathrm{bi}}}$ per mode, $(N-2)$ in total across both modes) and across modes (every inter-mode pair converges to the same distance, so it does not matter which single edge the minimum spanning tree uses to join the two modes; exactly one such edge is needed to connect the forest into a tree). Summing the intra-mode and the one inter-mode edge gives the stated formula.

Step 5 (Part C). By Step 2, standardization multiplies every edge of $\mathrm{MST}(\Delta)$ by the same scalar $\sigma\sqrt{D/\tau}$, which therefore cancels exactly from the ratio $b_l = \max_e w(e)/\mathcal{L}$ — this ratio depends only on the *relative* edge lengths established in Step 4. For the unimodal cloud all $(N-1)$ edges converge to the same length e, so the single longest edge is also length e and $b(\Delta_{\mathrm{uni}}) \to e/((N-1)e) = 1/(N-1)$. For the bimodal cloud, by Step 4, $\mathcal{L}(\Delta_{\mathrm{bi}})$ is the sum of $(N-2)$ bulk edges of length $\sqrt{2}\,R_w$ and one bridge edge of length $\sqrt{4\mu_\xi^2 + 2R_w^2} = \sqrt{2}\sqrt{2\mu_\xi^2 + R_w^2}$, and since $\mu_\xi > 0$ makes the bridge strictly the longest edge, $b(\Delta_{\mathrm{bi}}) \to \sqrt{2}\sqrt{2\mu_\xi^2 + R_w^2}\,/\,[(N-2)\sqrt{2}\,R_w + \sqrt{2}\sqrt{2\mu_\xi^2 + R_w^2}]$, which simplifies to the stated expression after canceling the common factor of $\sqrt{2}$. ■

O.4 Two regimes

The comparison in the Lemma is abstract; the following two concrete cases make it operational, and in both cases we track sum of norms alongside connectivity length rather than one at a time. The two cases differ only in what is held fixed between the unimodal and bimodal ξ: total scatter V_ξ in the first, the within-mode noise level R_w in the second.

Matched-scatter comparison ($R_u = \sqrt{R_w^2 + \mu_\xi^2}$, unimodal and bimodal ξ built to have identical aggregate scatter): $\tau_{\mathrm{uni}} = \tau_{\mathrm{bi}}$ exactly, so the dilution factor is 1 for *both* statistics, and each ratio reduces to its own shape factor alone. For connectivity length this leaves $R_w/\sqrt{R_w^2 + \mu_\xi^2} < 1$ as $N \to \infty$: the bimodal cloud is correctly scored lower. For sum of norms, Part A's shape factor is $\sqrt{V_{\xi,\mathrm{bi}}/V_{\xi,\mathrm{uni}}} = 1$ by construction (that is exactly what "matched scatter" means), so its ratio is exactly 1 — not wrong, but uninformative: it cannot distinguish the two clouds *by construction*, regardless of how differently their MSTs are organized. This is unaffected by μ_0, γ, since these cancel identically from τ_{uni} and τ_{bi} in either statistic once V_ξ is matched.

Matched-noise comparison ($R_u = R_w$, unimodal and bimodal share the same within-mode spread, and bimodal simply adds extra between-mode separation on top): now $\tau_{\mathrm{bi}} > \tau_{\mathrm{uni}}$, so *both* statistics' dilution factors drop below 1, but the two shape factors respond to this in opposite ways. Connectivity length's shape factor is itself > 1 for finite N (with $N = 40, R_w = 1, \mu_\xi = 1.5$: $[(N-2) + \kappa_{\mathrm{br}}]/(N-1) = (38 + 2.345)/39 \approx 1.034$, where $\kappa_{\mathrm{br}} := \sqrt{1 + 2\mu_\xi^2/R_w^2}$ is the bridge-to-bulk edge-length ratio, since one bridge edge $\kappa_{\mathrm{br}} \approx 2.35\times$ longer than typical outweighs saving one edge count out of 39), so the dilution factor is fighting against a shape factor that already points the wrong way; sum of norms' shape factor is $\sqrt{(R_w^2 + \mu_\xi^2)/R_w^2} \approx 1.803 > 1$, so its dilution factor is fighting against an even larger shape factor pointing the same wrong way. At $\mu_0 = \gamma = 0$, connectivity length's dilution factor is small enough (≈ 0.555) to win, giving a product $\approx 0.573 < 1$ — correctly ranking bimodal below unimodal, if only barely and by cancellation rather than by design. Sum of norms' dilution factor at $\mu_0 = \gamma = 0$ happens to invert to *exactly* $1/1.803$ (pooled standardization degenerates to displacement-cloud self-standardization at zero nuisance), so its product is also exactly 1 here — coincidentally blind rather than coincidentally correct.

As μ_0 (class separation) or γ (shared pair-noise) grow, both dilution factors rise toward 1, and each statistic's ratio drifts toward its own fixed shape factor: connectivity length's ratio **crosses above 1** near $\mu_0 \approx 5$ in our simulation, meaning it now ranks the mechanistically-split (bimodal) cloud *above* the coherent one; sum of norms' ratio climbs toward ≈ 1.80, moving even further from useful. So in the matched-noise comparison the two statistics fail in the *same direction* (both drift toward, and connectivity length crosses, the wrong ranking). For the *same reason* (dilution washing out as separation dominates τ) — connectivity length simply starts from a better position (a shape factor close to 1) and takes longer to fail. In comparison, sum of norms starts from a worse position (a shape factor of 1.8) and is wrong for nearly the entire sweep. Neither statistic's failure here is a coincidence of this specific R_w, μ_ξ choice: both trace back to the same dilution mechanism

identified in Step 2 of the proof, applied through two different shape factors. This identifies a risk to check for at well-separated layers specifically — it is a claim about the standardization pipeline, not a diagnosis of any particular row of Table 5, since that table does not report the separation, bridge share, or $\mathcal{L}$ values needed to confirm which mechanism, if any, is responsible for a given failure.

Bridge share b_l, by contrast, is unaffected by any of this in either regime: it depends only on $\tau_{\text{uni}} = \tau_{\text{bi}}$ or $\tau_{\text{uni}} < \tau_{\text{bi}}$ through a factor that cancels identically (Step 5), so it stays fixed at its two predicted values, $1/(N-1) \approx 0.026$ for unimodal and ≈ 0.058 for bimodal (with $N = 40, R_w = 1, \mu_\xi = 1.5$), regardless of μ_0, γ, or which comparison is used.

O.5 Simulation

We simulate the model above at $D = 500$, $N = 40$, $R_w = 1$, $\mu_\xi = 1.5$, $\sigma = 1$, $\gamma = 1$ (Prim's algorithm MST, mode-separation spread isotropically across all D coordinates, averaged over 8 trials per point), sweeping the class separation μ_0 from 0 to 12, under both the matched-scatter and matched-noise comparisons.

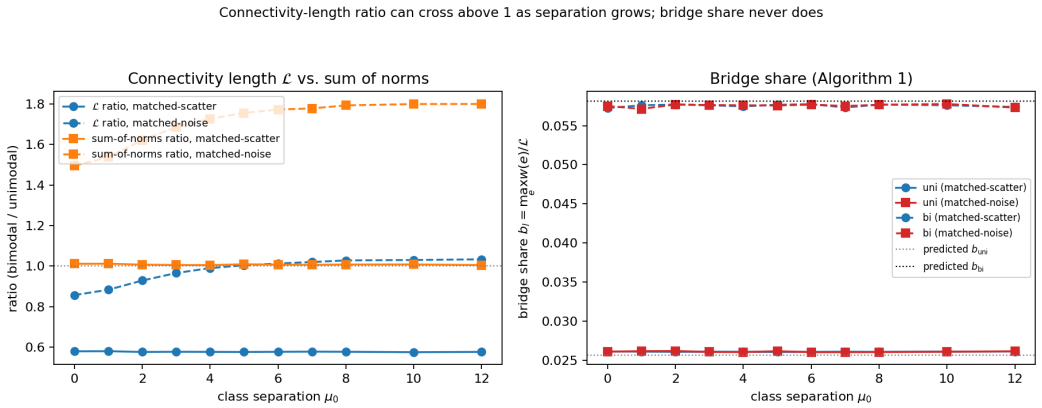

Figure 9: Left: connectivity-length ratio $\mathcal{L}(\Delta_{\text{bi}})/\mathcal{L}(\Delta_{\text{uni}})$ and sum-of-norms ratio, overlaid for direct comparison. The connectivity-length ratio (blue) is stable at ≈ 0.58 under the matched-scatter comparison (solid) across all μ_0, but under the matched-noise comparison (dashed) it rises with μ_0 and crosses above 1 near $\mu_0 \approx 5$, meaning the bimodal (mechanistically split) cloud is ranked *above* the unimodal one once class separation is large enough. The sum-of-norms ratio (orange) stays near 1 under matched-scatter (solid) but climbs toward ≈ 1.8 under the matched-noise comparison (dashed), becoming strongly shape-sensitive for a reason unrelated to representativeness. Right: the bridge share $b_l = \max_e w(e)/\mathcal{L}$ that Algorithm 1 already reports is essentially unchanged across every condition — ≈ 0.026 for unimodal, ≈ 0.058 for bimodal, matching the closed-form predictions of Part C exactly, regardless of μ_0 or comparison regime.

Table 12 reports selected values from the sweep underlying Figure 9.

Table 12: Selected points from the μ_0-sweep at $D = 500$, $\gamma = 1$. The connectivity-length ratio crosses above 1 in the matched-noise comparison as μ_0 grows, while the matched-scatter ratio and the bridge share (both regimes) stay essentially fixed at their predicted values.

μ_0	$\mathcal{L}$ ratio (m.s.)	$\mathcal{L}$ ratio (m.n.)	sn ratio (m.s.)	sn ratio (m.n.)	b_{uni} (m.n.)	b_{bi} (m.n.)
0	0.581	0.857	1.012	1.494	0.0261	0.0575
2	0.577	0.929	1.007	1.621	0.0262	0.0577
5	0.577	1.004	1.009	1.755	0.0262	0.0576
8	0.578	1.028	1.008	1.792	0.0261	0.0577
12	0.578	1.034	1.005	1.799	0.0262	0.0573

O.6 SUMMARY

Sum-of-norms is uninformative or actively misleading in every regime tested: it either cannot distinguish a coherent displacement cloud from a mechanistically split one, or does so only by leaking in class separation and shared noise that have nothing to do with representativeness. Connectivity length $\mathcal{L}$ is a genuine but conditional signal: it correctly favors the coherent layer when class separation is modest, but the same pooled standardization that TOPOSTEER relies on can let separation dominate the normalizing scale at well-separated layers, at which point $\mathcal{L}$ can rank a mechanistically split displacement cloud *above* a coherent one. Whether this occurs for any specific layer in Table 5 is an open empirical question — the table does not report the diagnostics (separation magnitude, bridge share) needed to check it, so we state this as a risk the theory identifies, not as an explanation already confirmed for any reported result. The bridge share b_l that Algorithm 1 already computes is unaffected by this: being a ratio of a single edge to the total, it is invariant to whatever scale the standardization imposes, and it separates unimodal from bimodal clouds by a fixed, predictable margin in every condition tested here. The practical recommendation is therefore to rank layers by $\mathcal{L}$ but to treat that ranking as valid only where b_l confirms it — not to read $\mathcal{L}$ magnitude alone as sufficient evidence of steerability.