Predictive Geometry of Hidden Trajectories in Transformers

Timur Mudarisov[1] **Mikhail Burtsev**[2] **Tatiana Petrova**[1] **Radu State**[1]
[1]University of Luxembourg, Luxembourg
[2]London Institute for Mathematical Sciences, London, UK

Abstract

Decoder-only transformers are trained only through a terminal next-token prediction loss, yet this loss constrains every intermediate hidden state through the fixed downstream computation. We formalize this constraint by studying layerwise loss-to-go functions: the terminal loss obtained by continuing a candidate hidden state through the remaining transformer blocks. Around successful validation trajectories, we show that the local second-order geometry of these functions is governed, up to low-loss residual terms, by a pullback Fisher operator on hidden-state space. Its spectrum identifies output-sensitive directions and approximately prediction-null directions, yielding a local observable subspace of the residual stream. For causal transformers, the same geometry induces a tokenwise curvature score: a Fisher-weighted sensitivity of the target logits to perturbations of each token's hidden state. This score vanishes outside the causal ancestor set of the target and is controlled by downstream Jacobian couplings, making it a loss-aware alternative to attention magnitude. We estimate these quantities using matrix-free Jacobian-vector and vector-Jacobian products and evaluate them across decoder-only language models on WikiText, OpenWebText, and FineWeb. Direct Hessian–vector comparisons confirm that the pullback Fisher term dominates the loss-to-go curvature along output-sensitive directions, and prediction-null directions remain nearly invisible to future token positions as well. Empirically, the induced geometry predicts perturbation sensitivity, supports nonuniform layerwise rank allocation, yields competitive structured token-pruning signals, and improves low-rank student recovery when added to stronger autoregressive distillation objectives such as reverse KL and skew KL. These results support a predictive-geometric view of transformer computation: near successful trajectories, the terminal loss induces a thin, anisotropic set of output-relevant hidden-state directions that can be measured and exploited for compression and distillation.

1 Introduction

Large-scale decoder-only transformers [25, 5] are optimized through a next-token prediction loss applied only at the output layer, yet their computation proceeds through a sequence of intermediate hidden states. This raises a local question about trained models: near hidden-state trajectories on which the model already predicts well, which directions in residual-stream space are constrained by the terminal prediction loss, and which directions are approximately invisible to it? Answering this question would give a loss-based account of hidden-state geometry, rather than treating intermediate representation geometry as an empirical property detached from the objective used to train the model.

Recent empirical work suggests that transformer hidden states are structured and compressible rather than generic high-dimensional vectors. Prior work [24] shows that the intrinsic dimension of hidden representations follows a characteristic expansion-contraction profile across layers, while

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

activation-space compression methods [3, 22, 28] demonstrate that low-rank approximations can often preserve model performance. These results indicate that hidden representations contain substantial redundancy. However, they largely characterize this structure empirically: they do not derive the relevant directions from the terminal prediction objective itself, nor do they provide a loss-aware notion of which hidden-state directions or token positions matter for the final prediction.

In this work, we start from the prediction loss. For a fixed trained decoder-only transformer, a layer, and a target position, we define a *loss-to-go* function: the terminal loss obtained by continuing a candidate intermediate hidden state through the fixed downstream transformer. We then study the local second-order geometry of this function around successful validation trajectories. In the low-loss regime, the dominant curvature is given by a pullback Fisher operator: the Fisher geometry of the output distribution pulled back through the downstream computation to the intermediate hidden state. This operator identifies output-sensitive directions in hidden-state space together with approximately prediction-null directions that leave the centered target logits unchanged to first order.

This local output-induced geometry leads to directly measurable quantities. The spectrum of the pullback Fisher operator measures how many output-sensitive directions are present at each layer, suggesting a data-dependent alternative to uniform rank allocation. Taking traces of this operator over token fibers defines a local token-level sensitivity score, measuring how much each token position contributes to the curvature of the target prediction. For causal transformers, this score is supported only on causal ancestors of the target token and is controlled by downstream Jacobian couplings, making it a loss-aware alternative to attention-based pruning heuristics [26, 17, 29]. We use these quantities as signals for matched-budget rank allocation and structured token pruning.

Our perspective connects representation geometry [24], compression and low-rank adaptation [3, 13, 11, 22], token pruning [26, 17, 29], mechanistic interpretability [6], and Fisher-based analyses of neural models [2, 15]. The focus here is specifically on the *local predictive geometry* of intermediate hidden states: the Fisher geometry induced by the terminal next-token loss and pulled back through the learned forward dynamics of a fixed trained transformer. This paper makes the following contributions:

1. **Loss-induced hidden geometry.** We define layerwise loss-to-go functions for fixed trained decoder-only transformers and show that, near successful hidden-state trajectories, their dominant local curvature is a pullback Fisher operator from the output distribution to the intermediate hidden state.
2. **Observable and prediction-null directions.** The pullback Fisher operator identifies output-sensitive directions and approximately prediction-null directions in residual-stream space, yielding a testable local decomposition of hidden-state perturbations, which we verify against exact Hessian–vector products and future-position sensitivities.
3. **Tokenwise predictive saliency.** For causal transformers, we derive a tokenwise Fisher-Jacobian score $\kappa_{\ell,s}$ that measures how much each token position contributes to target-logit curvature. The score is supported only on causal ancestors of the target and is controlled by downstream Jacobian couplings.
4. **Geometry-guided reduction and recovery.** We evaluate the resulting geometry across perturbation sensitivity, layerwise rank allocation, structured token pruning, and low-rank student distillation. Across these settings, the geometry provides loss-aware signals for preserving prediction-relevant hidden directions under compression.

2 Problem Statement

We study a fixed trained decoder-only pre-norm transformer at a fixed target position t. For an input prefix $s_{\le t}$, let

$$X_\ell(s_{\le t}) \in \mathcal{X}_t := \mathbb{R}^{t\times d}, \qquad \ell = 0, \dots, L,$$

denote the layer-ℓ hidden state, where d is the hidden dimension. The model induces a hidden trajectory

$$X_0(s_{\le t}) \to X_1(s_{\le t}) \to \cdots \to X_L(s_{\le t}),$$

and the final prediction at position t is read out from X_L. Writing the layer dynamics as

$$X_{\ell+1} = B_\ell(X_\ell), \qquad \Phi_{\ell\to L} := B_{L-1} \circ \cdots \circ B_\ell,$$

we analyze the geometry induced by the learned forward dynamics of a *fixed trained model*, rather than the training dynamics themselves.

To measure prediction quality at position t, let ϕ be a terminal loss applied after the final readout, such as cross-entropy or

$$\phi(X_L) := D_{\mathrm{KL}}\big(q \,\|\, p_t(\cdot \mid X_L)\big),$$

where $p_t(\cdot \mid X_L)$ is the prediction induced by the final hidden state X_L. Define the layerwise loss-to-go

$$\mathcal{J}_\ell(X) := \phi\big(\Phi_{\ell\to L}(X)\big),$$

and the associated low-loss set

$$\mathcal{A}_\varepsilon^\ell := \{\, X \in \mathcal{X}_t : \mathcal{J}_\ell(X) \le \varepsilon \,\}.$$

Thus, $\mathcal{A}_\varepsilon^\ell$ consists of those intermediate states at layer ℓ that can still evolve, under the fixed downstream transformer, to terminal loss at most ε.

Our goal is to characterize the geometry of these layerwise low-loss sets and its consequences for transformer computation. In particular, we ask whether successful trajectories remain inside a pullback family of low-loss regions, whether the local geometry of these regions is effectively lower-dimensional than the ambient residual space $\mathcal{X}_t$, and how this prediction-relevant geometry is distributed across token positions through the causal computation graph. Answering these questions gives both a trajectory-level description of transformer computation beyond the final logits and a basis for loss-aware model-reduction signals, including layerwise low-rank compression and token-sparse computation.

3 Theory

We study the local geometry of intermediate hidden states that remain compatible with low terminal prediction loss (Fig. 1). Throughout, we fix a trained decoder-only pre-norm transformer, a target position t, and a softmax readout, with terminal loss

$$\phi(z) := D_{\mathrm{KL}}\big(q \,\|\, \mathrm{softmax}(z)\big), \qquad z \in \mathbb{R}^{|\mathcal{V}|}.$$

Proofs are presented in the Appendix A.

3.1 Local geometry of layerwise low-loss sets

Let

$$X_{\ell+1} = B_\ell(X_\ell), \qquad \Phi_{\ell\to L} := B_{L-1} \circ \cdots \circ B_\ell, \qquad \Psi_\ell(X) := Z_t\big(\Phi_{\ell\to L}(X)\big).$$

Define the layerwise loss-to-go and low-loss set by

$$\mathcal{J}_\ell(X) := \phi\big(\Psi_\ell(X)\big), \qquad \mathcal{A}_\varepsilon^\ell := \{X \in \mathcal{X}_t : \mathcal{J}_\ell(X) \le \varepsilon\}. \tag{1}$$

Then

$$\mathcal{A}_\varepsilon^\ell = \Phi_{\ell\to L}^{-1}(\mathcal{A}_\varepsilon^L), \qquad \mathcal{A}_\varepsilon^L = \{X \in \mathcal{X}_t : \phi(Z_t(X)) \le \varepsilon\}, \tag{2}$$

so the main question is the *local geometry* of these sets near a successful hidden-state trajectory.

Theorem 1 (Local output geometry of low-loss sets)**.** *Let $X_\ell^* \in \mathcal{X}_t$ be a reference hidden state with*

$$\varepsilon_* := \mathcal{J}_\ell(X_\ell^*), \qquad \mathcal{J}_\ell(X) := \phi(\Psi_\ell(X)), \qquad \phi(z) := D_{\mathrm{KL}}(q \,\|\, \mathrm{softmax}(z))\,.$$

Assume that Ψ_ℓ is twice continuously differentiable in a neighborhood of X_ℓ^* and that

$$\left(\sum_{a=1}^{|\mathcal{V}|} \|\nabla^2 \Psi_{\ell,a}(X)\|_{\mathrm{op}}^2\right)^{1/2} \le M_\ell$$

in that neighborhood. For transformers with smooth activations and LayerNorm, this condition holds locally on neighborhoods where LayerNorm denominators are bounded away from zero and attention probabilities remain finite. Let

$$P := I - \tfrac{1}{|\mathcal{V}|}\mathbf{1}\mathbf{1}^\top, \qquad \bar{\Psi}_\ell := P\Psi_\ell, \qquad \bar{J}_\ell := D\bar{\Psi}_\ell(X_\ell^*),$$

and let

$$p^* := \mathrm{softmax}(\Psi_\ell(X_\ell^*)), \qquad F^* := \mathrm{diag}(p^*) - p^* p^{*\top}.$$

Then:

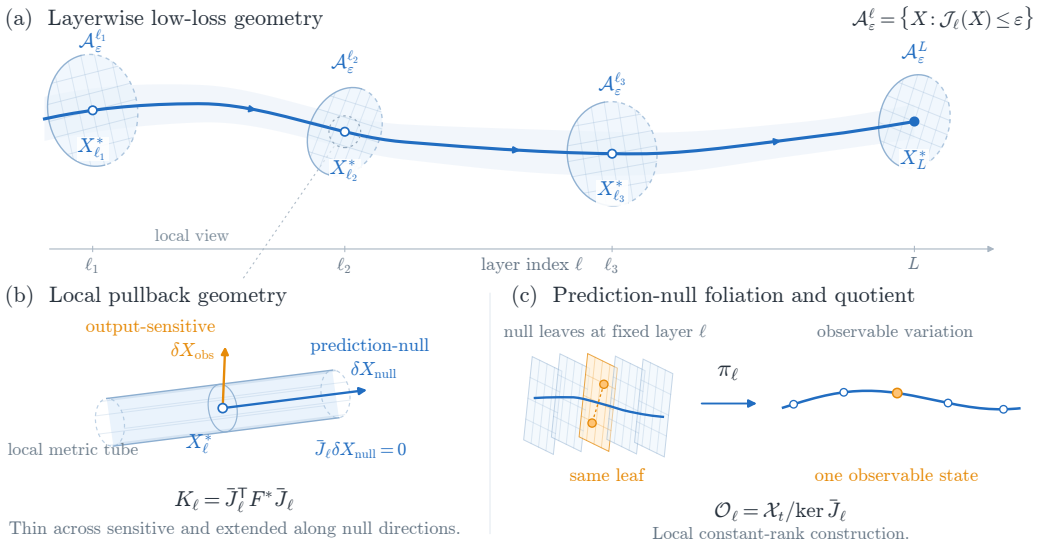

Figure 1: **Geometry of low-loss hidden trajectories.** **a**, Hidden-state trajectory and layerwise low-loss sets $\mathcal{A}_\varepsilon^\ell$. **b**, Local pullback geometry near X_ℓ^*. **c**, Prediction-null foliation and observable quotient.

1. *Local Hessian decomposition. The Hessian of the layerwise loss-to-go satisfies*

$$\nabla^2 \mathcal{J}_\ell(X_\ell^*) = \bar{J}_\ell^\top F^* \bar{J}_\ell + R_\ell, \tag{3}$$

where

$$\|R_\ell\|_{\text{op}} \le M_\ell \|p^* - q\|_2 \le M_\ell \sqrt{2\varepsilon_*}. \tag{4}$$

Thus, at low loss, the dominant curvature term is the pullback of the Fisher geometry of the output distribution.

2. *Pullback Fisher geometry. Define*

$$K_\ell := \bar{J}_\ell^\top F^* \bar{J}_\ell, \qquad g_\ell(\delta X, \delta X) := \langle \delta X, K_\ell \delta X \rangle. \tag{5}$$

For $X = X_\ell^* + \delta X$ sufficiently close to X_ℓ^*,

$$\mathcal{J}_\ell(X) = \mathcal{J}_\ell(X_\ell^*) + \langle \nabla \mathcal{J}_\ell(X_\ell^*), \delta X \rangle + \frac{1}{2} g_\ell(\delta X, \delta X) + O(\sqrt{\varepsilon_*} \|\delta X\|^2) + O(\|\delta X\|^3). \tag{6}$$

Hence, near a low-loss point with small first-order term, the local low-loss set is approximated by an ellipsoidal tube.

3. *Prediction-null directions. If $\bar{J}_\ell$ has constant rank r near X_ℓ^*, then*

$$\mathcal{N}_\ell := \ker \bar{J}_\ell \tag{7}$$

consists of first-order prediction-null directions.

Informal summary of Theorem 1

A successful hidden-state trajectory stays inside a "tube" of low-loss states. This tube is thin in prediction-relevant directions and thick in null directions; removing the latter leaves a smaller observable space whose local geometry is described by g_ℓ.

3.2 Tokenwise geometric saliency in transformers

Let

$$G_\ell := DB_\ell(X_\ell^*), \qquad G_\ell^{(r \leftarrow s)} : \mathbb{R}^d \to \mathbb{R}^d$$

denote the Jacobian of the ℓ-th block in token blocks. The operator K_ℓ in (5) is the local output-sensitive curvature operator. For each token s, let

$$P_s : \mathcal{X}_t \to \mathbb{R}^d$$

project onto the s-th token fiber, and define the geometric contribution score

$$\kappa_{\ell,s} := \operatorname{tr}(P_s K_\ell P_s^\top). \tag{8}$$

Thus $\kappa_{\ell,s}$ measures how much of the local output-sensitive curvature is carried by token s.

For each token pair (s, r), define

$$w_\ell^{(r\leftarrow s)} := \|G_\ell^{(r\leftarrow s)}\|_F.$$

These weights induce a causal directed token graph at layer ℓ. Let $\mathrm{Anc}_{\ell:L-1}(t)$ be the ancestor closure of the target token t under the composed layerwise graphs.

Proposition 1 (Tokenwise pullback saliency and causal support)**.** *Let $K_\ell = \bar{J}_\ell^\top F^* \bar{J}_\ell$ be the pullback Fisher operator from Theorem 1. For a token position $s \le t$, let $P_s : \mathcal{X}_t \to \mathbb{R}^d$ denote projection onto the s-th token fiber, and define the downstream token-to-logit Jacobian*

$$H_{\ell,s} := D_{X_{\ell,s}} \bar{\Psi}_\ell(X_\ell^*) = \bar{J}_\ell P_s^\top \in \mathbb{R}^{|\mathcal{V}|\times d}.$$

Define the tokenwise curvature score

$$\kappa_{\ell,s} := \operatorname{tr}(P_s K_\ell P_s^\top).$$

Then

1. *Fisher-Jacobian identity.*

$$\kappa_{\ell,s} = \operatorname{tr}\big(H_{\ell,s}^\top F^* H_{\ell,s}\big) = \Big\| F^{*1/2} H_{\ell,s} \Big\|_F^2. \tag{9}$$

   Thus $\kappa_{\ell,s}$ is the Fisher-weighted squared sensitivity of the target-position logits to perturbations of token s at layer ℓ.

2. *Causal support. Under the standard causal mask, if token s is not in the downstream ancestor set of target token t from layer ℓ to L, then*

$$H_{\ell,s} = 0, \qquad \kappa_{\ell,s} = 0.$$

3. *Pathwise coupling bound. Let $G_m^{(r\leftarrow u)}$ denote the token-block Jacobian of block m, mapping perturbations at token u to perturbations at token r. Let $\Pi_{\ell:s\to t}$ be the set of directed causal paths from (ℓ, s) to the target token t at layer L. Then*

$$\kappa_{\ell,s} \le d\|F^*\|_{\mathrm{op}} \|DZ_t(X_L^*)\|_{\mathrm{op}}^2 \left(\sum_{\pi\in\Pi_{\ell:s\to t}} \prod_{(m,u\to r)\in\pi} \|G_m^{(r\leftarrow u)}\|_{\mathrm{op}} \right)^2. \tag{10}$$

Informal summary of Proposition 1

A token contributes to the output geometry only if it causally influences the target. Even among such ancestors, its contribution is small when its Jacobian coupling to the target is weak. Geometric importance is therefore both sparse and graded.

3.3 Geometry-guided reduction

The spectrum of K_ℓ identifies prediction-relevant directions, while Proposition 1 shows that this geometry is often concentrated on a small subset of ancestor tokens. This suggests projecting onto the dominant eigenspace of K_ℓ and pruning tokens with small $\kappa_{\ell,s}$.

Proposition 2 (Geometry-guided reduction with local loss guarantee)**.** *Let K_ℓ be the pullback Fisher operator from Theorem 1, with eigenvalues*

$$\lambda_1(K_\ell) \geq \lambda_2(K_\ell) \geq \cdots \geq 0.$$

Let $\Pi_\ell^{(k)}$ be the orthogonal projector onto the span of the top k eigenvectors of K_ℓ. For a local hidden-state perturbation

$$X = X_\ell^* + \delta X,$$

define its geometry-guided reduction by

$$\widetilde{X} := X_\ell^* + \Pi_\ell^{(k)} \delta X.$$

Assume that X and $\widetilde{X}$ lie in the neighborhood where Theorem 1 applies. Then the discarded component

$$\delta X_{\mathrm{tail}} := (I - \Pi_\ell^{(k)})\delta X$$

has local quadratic loss contribution bounded by

$$\frac{1}{2}\langle \delta X_{\mathrm{tail}}, K_\ell \delta X_{\mathrm{tail}} \rangle \leq \frac{1}{2}\lambda_{k+1}(K_\ell)\|\delta X_{\mathrm{tail}}\|^2. \tag{11}$$

4 Empirical Validation and Geometry-Guided Reduction

The theory yields two kinds of objects. The first are *descriptive* observables of the hidden trajectory, such as the operator K_ℓ and the tokenwise score $\kappa_{\ell,s}$, which can be measured directly on trained models and used to test the predictions of Theorem 1 and Proposition 1. The second are *constructive* objects, such as layerwise observable subspaces and tokenwise saliency rankings, which can be used to design reduced architectures and geometry-aware compression schemes.

This section evaluates the theory at two levels. First, we test whether the pullback-Fisher geometry is a faithful local description of hidden-state sensitivity: observable directions should be substantially more output-sensitive than prediction-null directions. Second, we test whether the same geometry is useful as an algorithmic signal for compression and recovery. **Experiment 0** (Section 4.1) directly validates the local Taylor geometry induced by the pullback Fisher operator. **Experiment 1** (Section 4.2) tests the observable/null anisotropy and effective observable dimension predicted by the low-loss tube picture, and **Experiment 1b** (Section 4.3) checks it against exact Hessian–vector products and future-position sensitivity. **Experiment 2** (Section 4.4) uses the spectrum of K_ℓ for matched-budget rank allocation. **Experiment 3** (Section 4.5) uses the token-fiber trace $\kappa_{\ell,s}$ for structured token-update pruning. **Experiment 4** (Section 4.6) uses the same geometry as a hidden-state recovery objective for low-rank student distillation.

Unless stated otherwise, all measurements are computed on held-out validation sequences from WIKITEXT [18], FINEWEB [21], and OPENWEBTEXT [8].

4.1 Experiment 0: Direct validation of local predictive geometry

Before using the pullback Fisher operator for reduction or recovery, we first test whether it captures the local predictive geometry predicted by Theorem 1. Specifically, we ask whether small perturbations of an intermediate hidden state produce terminal loss changes predicted by the local Taylor model induced by K_ℓ. For supervised cross-entropy, we compare a linear predictor, a Fisher quadratic predictor, and their sum: $\Delta\mathrm{CE}_{\mathrm{lin}} = \langle \nabla_{X_\ell}\mathrm{CE}, \delta X_\ell \rangle$, $\Delta\mathrm{CE}_{\mathrm{quad}} = \frac{1}{2}\delta X_\ell^\top K_\ell \delta X_\ell$, and $\Delta\mathrm{CE}_{\mathrm{lin+quad}} = \Delta\mathrm{CE}_{\mathrm{lin}} + \Delta\mathrm{CE}_{\mathrm{quad}}$. This separation is necessary because supervised one-hot CE is generally not stationary at the reference prediction, so the linear term need not vanish. To isolate the Fisher curvature term, we also evaluate teacher-KL with $q = p^\star$, the model's unperturbed prediction. In this stationary setting, the leading local term is $\Delta\mathrm{KL}(p^\star \| p_\delta) \approx \frac{1}{2}\delta X_\ell^\top K_\ell \delta X_\ell$.

Results show that supervised CE is first-order dominated: the linear-only and linear-plus-quadratic predictors nearly coincide, while the quadratic-only predictor is substantially weaker (see Fig. 4 in App. D.1). In contrast, the teacher-KL quadratic predictor becomes strongly predictive at moderate radii, where the perturbation-induced KL signal is measurable. Thus, Experiment 0 validates K_ℓ as a local output-curvature or observability operator.

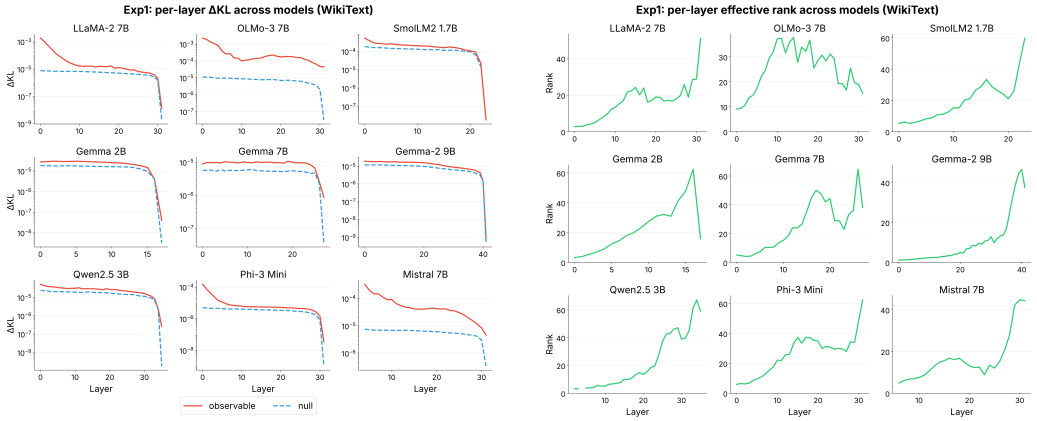

Figure 2: **Layerwise output anisotropy and effective observable dimension across decoder-only LLMs. Left:** Target-position output sensitivity (ΔKL) resulting from equal-norm perturbations sampled along top observable eigenspaces (solid red) versus prediction-null directions (dashed blue) across layers on WIKITEXT. Observable perturbations produce orders-of-magnitude higher output disruption than null perturbations at almost every layer, confirming strong local anisotropy. **Right:** Per-layer effective observable rank r_{eff}. Across all nine evaluated models, r_{eff} expands gradually with layer depth as representations format toward the terminal readout, yet remains orders of magnitude smaller than the ambient hidden dimension ($r_{\text{eff}} \ll d$).

4.2 Experiment 1: anisotropy and effective observable dimension

Guided by Theorem 1, we quantify how strongly each layer reacts to perturbations in observable versus prediction-null directions. For every model, layer ℓ, and target position t, we construct the local curvature operator K_ℓ, compute its top eigendirections, and use them to define an observable subspace together with its orthogonal complement. We then sample equal-norm perturbations in each subspace, apply them to $X_\ell(s)$, and measure the resulting change in target-position KL divergence between the original and perturbed outputs.

To summarize the spectrum of K_ℓ, we report the effective observable rank via the participation ratio

$$r_{\text{eff}}(K_\ell) := \frac{\left(\sum_i \lambda_i(K_\ell)\right)^2}{\sum_i \lambda_i(K_\ell)^2}. \tag{12}$$

This quantity serves as an empirical proxy for the low-dimensional prediction-relevant structure suggested by Theorem 1. Figure 2 shows a clear and consistent anisotropy across all models: observable perturbations are orders of magnitude more harmful than null perturbations at almost every layer, and the gap typically widens toward the later blocks. This indicates that the residual geometry becomes increasingly aligned with the final readout as the computation approaches the output layer. The right panel further shows that the effective observable rank grows gradually with depth, especially in the middle and later layers, but remains far below the ambient hidden dimension. An independent finite-difference estimate of r_{eff} (Section 4.3) confirms the magnitude. Taken together, these trends support the view that decoder-only transformers concentrate most of their prediction-relevant information in a thin, highly anisotropic subset of hidden space, while a large fraction of directions remain nearly KL-neutral.

4.3 Experiment 1b: direct curvature and future-position checks

Experiments 0–1 rest on two further premises. First, the Fisher term should dominate the full loss-to-go Hessian along output-sensitive directions, i.e., the residual R_ℓ in (3) should be small there. Second, directions that are null for the target t should not merely move sensitivity to later positions $t' > t$, which also read the perturbed hidden state through causal attention. We test both on LLaMA-2-7B and Mistral-7B (all 32 layers, OPENWEBTEXT, radius $\rho = 10^{-3}$). For the first, we compare exact supervised-CE Hessian–vector products $H_{\text{CE}}v = \nabla^2 \mathcal{J}_\ell(X_\ell^\star)v$ with $K_\ell v$, and $v^\top K_\ell v$ with the centered finite-difference teacher-KL curvature. For the second, we form the horizon operator

$K_\ell^H = \sum_{t' \in H} K_{\ell,t'}$, where $K_{\ell,t'}$ is (5) for target t', over a window H of subsequent positions and measure the summed KL curvature at future positions. Besides projected-null and random directions, we probe *data-driven* directions: the null-space component of real layer updates and of inter-example hidden-state differences (details in Appendix G).

Table 1: **Pullback Fisher operator aligns with exact Hessian curvature while output sensitivity occupies a low-dimensional, future-isolated subspace. Left:** Agreement between exact supervised-CE Hessian–vector products ($\nabla^2 \mathcal{J}_\ell v$) and pullback Fisher products ($K_\ell v$) on top-Fisher directions, alongside Spearman rank correlation between $v^\top K_\ell v$ and finite-difference teacher-KL curvature. **Middle:** Median KL curvature of single-target null directions relative to top directions at the target position and summed across future context positions ($t' > t$); "layer upd." denotes data-driven null perturbations from actual block updates. **Right:** Mean effective observable rank r_{eff} estimated via finite-difference (FD) moments and matrix-free Hutchinson probes.

	Hessian vs. Fisher		Null / top KL curvature			r_{eff}
Model	cos ↑	Spearman ↑	target	future	layer upd. (future)	FD / Hutch.
LLaMA-2-7B	0.87	0.98	$1.1 \cdot 10^{-3}$	$2.6 \cdot 10^{-3}$	$2.6 \cdot 10^{-3}$	22.7 / 10.1
Mistral-7B	0.64	0.98	$4.9 \cdot 10^{-4}$	$1.8 \cdot 10^{-3}$	$9.6 \cdot 10^{-4}$	30.3 / 33.2

Table 1 supports both premises. Along top-Fisher directions, $H_{\text{CE}} v$ and $K_\ell v$ are well aligned and $v^\top K_\ell v$ ranks the measured teacher-KL curvature almost perfectly. Alignment is weakest in the first layers of Mistral-7B and reaches ≈ 0.95 near the readout (App. G, Fig. 8), consistent with a residual R_ℓ that grows with distance from the output. Along null directions, both quadratic forms drop by five to six orders of magnitude, so the large relative Hessian–Fisher mismatch there reflects a residual at the numerical floor rather than hidden curvature. Single-target null directions do leak to future positions (their future-to-immediate ratio is 3.7 and 8.0), but their absolute future effect stays below 0.3% of that of horizon-top directions and matches random directions. Real layer updates and inter-example differences behave the same way. The loss-aligned CE gradient is a stress case: only 0.4% (LLaMA) and 15% (Mistral) of its energy lies in the estimated null space, and even that component retains 10–18% of top-direction sensitivity, so the null space does not trivially absorb arbitrary directions.

4.4 Experiment 2: geometry-guided rank allocation

Experiment 2 asks whether the layerwise geometry can be used to allocate low rank more effectively than uniform compression. Given a global compression budget, our geometry-guided method assigns a data-dependent rank k_ℓ to each layer using the spectrum of K_ℓ. Concretely, for a threshold $\tau \in \{0.01, 0.05, 0.1, 0.2\}$, we choose the smallest k_ℓ such that

$$\sum_{i > k_\ell} \lambda_i(K_\ell) \;\le\; \tau \sum_i \lambda_i(K_\ell). \tag{13}$$

Intuitively, τ controls how much spectral tail mass is discarded. As controlled matched-budget baselines, we use uniform SVD and uniform LoRA-style low-rank allocations [13, 11], assigning the same rank to every layer and matching the overall compression ratio.

For each model, dataset, and threshold τ, we report the perplexity increase

$$\Delta\text{PPL} = \text{PPL}_{\text{compressed}} - \text{PPL}_{\text{base}} \tag{14}$$

and the corresponding advantage over the best matched uniform baseline,

$$\Delta\text{PPL}_{\text{adv}} := \Delta\text{PPL}_{\text{uniform}} - \Delta\text{PPL}_{\text{geom}}. \tag{15}$$

Larger $\Delta\text{PPL}_{\text{adv}}$ therefore indicates a stronger benefit from geometry-guided allocation. We evaluate ten decoder-only LLMs between roughly 1B and 9B parameters (see Table 3).

Figure 3 shows that geometry-guided rank allocation is most helpful at moderate thresholds ($\tau = 0.05$ and $\tau = 0.1$), where several small and medium models, including Gemma 2B, Qwen2.5 3B, SmolLM2 1.7B, and OLMo-3 7B, benefit from a nonuniform rank distribution. This suggests that these architectures contain a mix of strongly prediction-relevant and comparatively redundant layers. For larger and more homogeneous model families, especially some LLaMA and Gemma variants, the advantage is smaller, indicating that uniform low-rank compression is already close to optimal when layerwise spectra look similar.

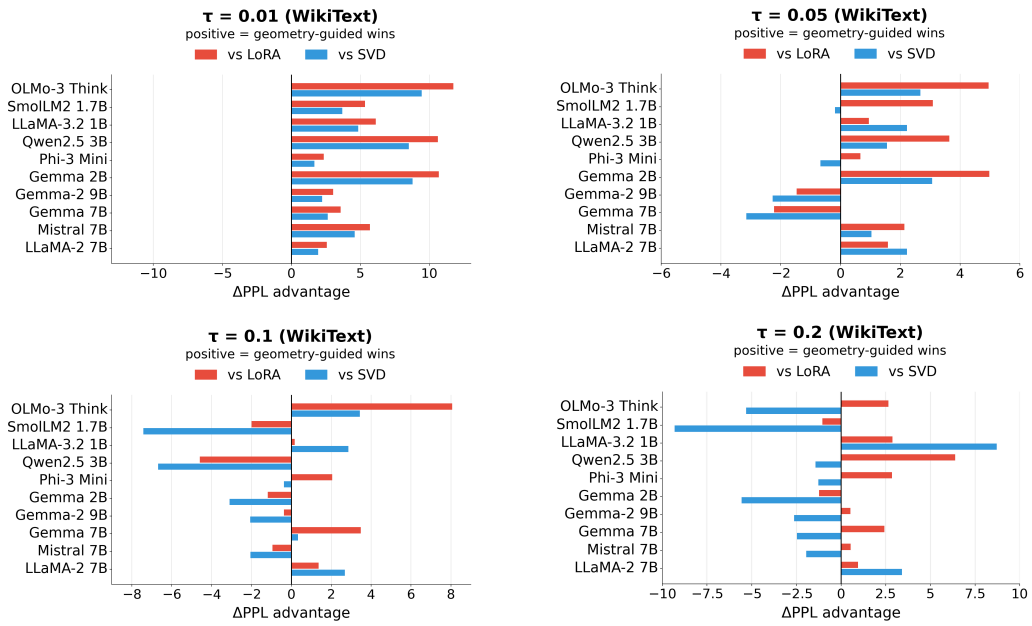

Figure 3: **Geometry-guided rank allocation outperforms uniform LoRA and SVD baselines under matched compression budgets.** Perplexity advantage $\Delta\text{PPL}_{\text{adv}} = \Delta\text{PPL}_{\text{uniform}} - \Delta\text{PPL}_{\text{geom}}$ on WIKITEXT across ten decoder-only language models for four spectral tail thresholds $\tau \in \{0.01, 0.05, 0.1, 0.2\}$. Layerwise ranks k_ℓ are chosen non-uniformly so that the discarded spectral tail mass of the pullback Fisher operator satisfies $\sum_{i>k_\ell} \lambda_i(K_\ell) \le \tau \sum_i \lambda_i(K_\ell)$. Positive values ($\Delta\text{PPL}_{\text{adv}} > 0$) indicate that data-dependent rank allocation according to K_ℓ degrades perplexity less than parameter-matched uniform LoRA (red) or uniform SVD (blue) baselines.

4.5 Experiment 3: geometry-guided pruning

Experiment 3 studies whether the same geometric signals can guide structured pruning of layer updates. Starting from a compressed model, we rank tokenwise updates within each layer according to four policies: (i) the geometric saliency score $\kappa_{\ell,s}$, (ii) an attention-score baseline, (iii) a gradient-activation saliency baseline $\|\nabla_{X_{\ell,s}} \mathcal{L} \odot X_{\ell,s}\|_1$, and (iv) a random baseline. For a drop fraction $\rho \in \{0.05, 0.10, 0.20, 0.30\}$, we prune the least important fraction of token-indexed updates under each policy and measure the resulting validation perplexity increase

$$\Delta\text{PPL} = \text{PPL}_{\text{pruned}} - \text{PPL}_{\text{unpruned}}, \tag{16}$$

where lower is better. We run this experiment on the same family of decoder-only transformers as in Experiment 2 across WIKITEXT, OPENWEBTEXT, and FINEWEB. Results show that both structured criteria, $\kappa_{\ell,s}$ and attention score, substantially outperform random pruning across drop fractions (Fig. 6, Appp. F). The κ-based policy is competitive with attention-based pruning and is slightly better in several model/drop-fraction settings. This suggests that the geometric score captures a loss-aware notion of token importance while attention remains a strong baseline for the pruning.

4.6 Experiment 4: predictive-geometry distillation of low-rank students

The preceding experiments test predictive geometry as a static signal. We now ask a stronger question: can the same geometry improve *training* after compression?

Given a teacher T, we construct a compressed student S by replacing selected attention and MLP projections with rank-r low-rank factors,

$$W \approx AB, \qquad A \in \mathbb{R}^{d_{\text{out}} \times r}, \qquad B \in \mathbb{R}^{r \times d_{\text{in}}}. \tag{17}$$

The student is trained with an output-level KD objective [12] and, optionally, a predictive-geometry regularizer on hidden states. The geometry term penalizes student errors according to their teacher-predictive effect rather than their Euclidean size: errors in output-sensitive directions are penalized more heavily, while errors in approximately prediction-null directions are allowed to be larger.

We separate the role of compression from the role of distillation by comparing two initializers, SVD and activation-aware SVD, and three KD objectives: forward KL [12], reverse KL [10], and skew KL [16]. For each KD objective, we evaluate both the base variant and its geometry-regularized counterpart. The main paired comparisons are therefore RKL versus RKL+Geo and SkewKL versus SkewKL+Geo at matched model, rank, seed, and initializer. More details are in App. E.

Table 2: Geometry advantage over matched KD baselines. $\Delta_{\mathrm{Geo}} = \mathrm{PPL}(\mathrm{KD}) - \mathrm{PPL}(\mathrm{KD} + \mathrm{Geo})$; higher is better.

KD baseline	Pairs	Wins	Mean $\Delta_{\mathrm{Geo}} \uparrow$	Median $\Delta_{\mathrm{Geo}} \uparrow$	SVD mean	ASVD mean
FKL	14	8	-38.4	-12.6	-35.0	-41.7
RKL	14	9	21.3	12.3	44.5	-1.9
SkewKL	14	10	59.3	8.6	66.7	51.9

The geometry term is most useful when added to stronger autoregressive KD objectives. While adding geometry to forward KL does not improve average PPL, it improves reverse KL in 9/14 matched settings and skew KL in 10/14 matched settings. The largest average gain is obtained for skew KL, suggesting that predictive geometry is complementary to mode-seeking or skewed KD objectives rather than simply replacing output-level distillation.

5 Conclusion

We introduced a local Fisher-geometric view of hidden trajectories in decoder-only transformers. Starting from layerwise loss-to-go sets, we showed that, near successful validation trajectories and in the low-loss regime, the terminal prediction loss induces a pullback Fisher operator K_ℓ on intermediate hidden states. This operator separates output-sensitive directions from approximately prediction-null directions, yielding a local observable structure inside the residual stream.

For causal transformers, the same geometry induces a tokenwise curvature score $\kappa_{\ell,s}$, which measures the contribution of each token position to local output-sensitive curvature and vanishes outside the causal ancestor set of the target. Together, the spectrum of K_ℓ and its tokenwise traces provide loss-aware signals for where computation is concentrated, which directions should be preserved, and which components can be compressed with limited predictive damage.

Empirically, the proposed geometry gives a coherent and useful picture across multiple reduction settings. It predicts perturbation sensitivity, supports matched-budget layerwise rank allocation, yields competitive structured pruning signals, and provides an effective regularizer for low-rank student recovery when combined with stronger autoregressive distillation objectives. These results suggest that predictive geometry is not merely a descriptive diagnostic, but a practical mechanism for identifying and preserving the parts of the residual stream that matter for the model's predictions.

Overall, our findings support the view that transformer computation near successful trajectories is organized around a thin, anisotropic set of prediction-relevant directions induced by the terminal loss. Measuring this geometry turns the final prediction objective into actionable local information inside the network, opening a path toward geometry-aware compression, distillation, and efficient inference methods that are directly tied to the model's predictive behavior.

Acknowledgments and Disclosure of Funding

We thank Nebius for providing the computational resources used in this work. We also thank the anonymous reviewers for their constructive feedback, which substantially improved the paper.

References

[1] M. Abdin et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

[2] S.-i. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998.

[3] S. Ashkboos, M. L. Croci, M. G. d. Nascimento, T. Hoefler, and J. Hensman. SliceGPT: Compress large language models by deleting rows and columns. In *International Conference on Learning Representations*, 2024.

[4] L. Ben Allal, A. Lozhkov, E. Bakouch, G. Martín Blázquez, G. Penedo, L. Tunstall, A. Marafioti, H. Kydlíček, A. Piqueres Lajarín, V. Srivastav, J. Lochner, C. Fahlgren, X.-S. Nguyen, C. Fourrier, B. Burtenshaw, H. Larcher, H. Zhao, C. Zakka, M. Morlon, C. Raffel, L. von Werra, and T. Wolf. SmolLM2: When smol goes big—data-centric training of a small language model. *arXiv preprint arXiv:2502.02737*, 2025.

[5] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.

[6] N. Elhage, N. Nanda, C. Olsson, T. Henighan, N. Joseph, B. Mann, A. Askell, Y. Bai, A. Chen, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

[7] Gemma Team. Gemma: Open models based on Gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

[8] A. Gokaslan, V. Cohen, E. Pavlick, and S. Tellex. OpenWebText Corpus. `https://skylion007.github.io/OpenWebTextCorpus/`, 2019. Open-source replication of the WebText corpus.

[9] A. Grattafiori et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[10] Y. Gu, L. Dong, F. Wei, and M. Huang. MiniLLM: Knowledge distillation of large language models. In *International Conference on Learning Representations*, 2024.

[11] S. Hayou, N. Ghosh, and B. Yu. LoRA+: Efficient low rank adaptation of large models. In *International Conference on Machine Learning*, 2024.

[12] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. In *NeurIPS Deep Learning and Representation Learning Workshop*, 2015.

[13] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-rank adaptation of large language models. *International Conference on Learning Representations*, 2022.

[14] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. Le Scao, T. Lavril, T. Wang, T. Lacroix, and W. El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

[15] R. Karakida, S. Akaho, and S.-i. Amari. Understanding approximate fisher information for fast convergence of natural gradient descent in wide neural networks. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

[16] J. Ko, S. Kim, T. Chen, and S.-Y. Yun. DistiLLM: Towards streamlined distillation for large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 24872–24895. PMLR, 2024.

[17] J. Li, L. L. Zhang, J. Xu, Y. Wang, S. Yan, Y. Xia, Y. Yang, T. Cao, H. Sun, W. Deng, Q. Zhang, and M. Yang. Constraint-aware and ranking-distilled token pruning for efficient transformer inference. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023.

[18] S. Merity, C. Xiong, J. Bradbury, and R. Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.

[19] OLMo Team. OLMo 3. *arXiv preprint arXiv:2512.13961*, 2025.

[20] B. A. Pearlmutter. Fast exact multiplication by the hessian. *Neural Computation*, 6(1):147–160, 1994.

[21] G. Penedo, H. Kydlíček, L. Ben Allal, A. Lozhkov, M. Mitchell, C. Raffel, L. von Werra, and T. Wolf. The FineWeb datasets: Decanting the web for the finest text data at scale. *arXiv preprint arXiv:2406.17557*, 2024.

[22] J. Tian et al. FLAT-LLM: Fine-grained low-rank activation space transformation for large language model compression. In *Findings of the Association for Computational Linguistics: EACL*, 2026.

[23] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[24] L. Valeriani, D. Doimo, F. Cuturello, A. Laio, A. Ansuini, and A. Cazzaniga. The geometry of hidden representations of large transformer models. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[26] H. Wang, B. Dedhia, and N. K. Jha. Zero-tprune: Zero-shot token pruning through leveraging of the attention graph in pre-trained transformers. 2023.

[27] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.

[28] Z. Yuan, Y. Shang, Y. Song, Q. Wu, Y. Yan, and G. Sun. ASVD: Activation-aware singular value decomposition for compressing large language models. *arXiv preprint arXiv:2312.05821*, 2023.

[29] J. Yun, M. Kim, and Y. Kim. Focus on the core: Efficient attention via pruned token compression. In *Findings of the Association for Computational Linguistics: EMNLP*, 2023.

A Proofs

This appendix gives the proofs of the theoretical results stated in Section 3. We use the same notation as in the main text. In particular, $\Psi_\ell : \mathcal{X}_t \to \mathbb{R}^{|\mathcal{V}|}$ denotes the downstream logit map from layer ℓ to the target-position logits, and

$$\mathcal{J}_\ell(X) = \phi(\Psi_\ell(X)), \qquad \phi(z) = D_{\mathrm{KL}}(q \,\|\, \mathrm{softmax}(z))\,.$$

We write $p(z) = \mathrm{softmax}(z)$. The centering projection

$$P = I - \frac{1}{|\mathcal{V}|}\mathbf{1}\mathbf{1}^\top$$

removes the logit-shift direction, which is invisible to the softmax.

A.1 Preliminaries: softmax Fisher geometry

We first record elementary identities for the KL loss on logits.

Lemma 1 (Softmax Fisher identities)**.** *Let*

$$\phi(z) = D_{\mathrm{KL}}(q \,\|\, \mathrm{softmax}(z))\,, \qquad p = \mathrm{softmax}(z).$$

Then

$$\nabla_z \phi(z) = p - q, \qquad \nabla_z^2 \phi(z) = F(p) := \mathrm{diag}(p) - pp^\top.$$

Moreover,

$$F(p)\mathbf{1} = 0,$$

so the Hessian is invariant to additive shifts of the logits.

Proof. Since

$$\phi(z) = \sum_a q_a \log q_a - \sum_a q_a \log p_a(z),$$

and

$$\log p_a(z) = z_a - \log \sum_b e^{z_b},$$

we have

$$\phi(z) = \mathrm{const} - \sum_a q_a z_a + \log \sum_b e^{z_b}.$$

Therefore

$$\nabla_z \phi(z) = -q + p.$$

Differentiating $p = \mathrm{softmax}(z)$ gives

$$\frac{\partial p_a}{\partial z_b} = p_a(\delta_{ab} - p_b),$$

and hence

$$\nabla_z^2 \phi(z) = \mathrm{diag}(p) - pp^\top.$$

Finally,

$$F(p)\mathbf{1} = \mathrm{diag}(p)\mathbf{1} - pp^\top\mathbf{1} = p - p = 0.$$

□

A.2 Proof of Theorem 1

Proof. We prove the three claims in order.

1. Local Hessian decomposition. Let

$$\bar{\Psi}_\ell = P\Psi_\ell, \qquad \bar{J}_\ell = D\bar{\Psi}_\ell(X_\ell^*).$$

Because the softmax is invariant to additive constants,

$$\mathrm{softmax}(\Psi_\ell(X)) = \mathrm{softmax}(\bar{\Psi}_\ell(X))$$

up to the harmless choice of centered logit representative. Thus we may differentiate through $\bar{\Psi}_\ell$.

By the second-order chain rule,

$$\nabla^2 \mathcal{J}_\ell(X) = D\bar{\Psi}_\ell(X)^\top \nabla_z^2 \phi(\bar{\Psi}_\ell(X)) D\bar{\Psi}_\ell(X) + \sum_{a=1}^{|\mathcal{V}|} \frac{\partial \phi}{\partial z_a}(\bar{\Psi}_\ell(X)) \nabla^2 \bar{\Psi}_{\ell,a}(X).$$

Evaluating at $X = X_\ell^*$, using Lemma 1, and writing

$$p^* = \mathrm{softmax}(\Psi_\ell(X_\ell^*)), \qquad F^* = \mathrm{diag}(p^*) - p^* p^{*\top},$$

we obtain

$$\nabla^2 \mathcal{J}_\ell(X_\ell^*) = \bar{J}_\ell^\top F^* \bar{J}_\ell + R_\ell,$$

where

$$R_\ell = \sum_{a=1}^{|\mathcal{V}|} (p_a^* - q_a) \nabla^2 \bar{\Psi}_{\ell,a}(X_\ell^*).$$

By Cauchy-Schwarz,

$$\|R_\ell\|_{\mathrm{op}} \le \left(\sum_{a=1}^{|\mathcal{V}|} (p_a^* - q_a)^2 \right)^{1/2} \left(\sum_{a=1}^{|\mathcal{V}|} \|\nabla^2 \bar{\Psi}_{\ell,a}(X_\ell^*)\|_{\mathrm{op}}^2 \right)^{1/2}.$$

The second factor is bounded by M_ℓ by assumption, hence

$$\|R_\ell\|_{\mathrm{op}} \le M_\ell \|p^* - q\|_2.$$

Since $\|u\|_2 \le \|u\|_1$, and Pinsker's inequality gives

$$\|p^* - q\|_1 \le \sqrt{2 D_{\mathrm{KL}}(q\|p^*)} = \sqrt{2\varepsilon_*},$$

we get

$$\|R_\ell\|_{\mathrm{op}} \le M_\ell \sqrt{2\varepsilon_*}.$$

2. Pullback Fisher geometry. Define

$$K_\ell = \bar{J}_\ell^\top F^* \bar{J}_\ell.$$

By Taylor expansion of $\mathcal{J}_\ell$ around X_ℓ^*, for $X = X_\ell^* + \delta X$,

$$\mathcal{J}_\ell(X_\ell^* + \delta X) = \mathcal{J}_\ell(X_\ell^*) + \langle \nabla \mathcal{J}_\ell(X_\ell^*), \delta X \rangle + \frac{1}{2} \langle \delta X, \nabla^2 \mathcal{J}_\ell(X_\ell^*) \delta X \rangle + O(\|\delta X\|^3),$$

provided the third derivative is bounded in the neighborhood. Substituting the decomposition above gives

$$\mathcal{J}_\ell(X_\ell^* + \delta X) = \mathcal{J}_\ell(X_\ell^*) + \langle \nabla \mathcal{J}_\ell(X_\ell^*), \delta X \rangle + \frac{1}{2} \langle \delta X, K_\ell \delta X \rangle + \frac{1}{2} \langle \delta X, R_\ell \delta X \rangle + O(\|\delta X\|^3).$$

Using the bound on R_ℓ,

$$|\langle \delta X, R_\ell \delta X \rangle| \le M_\ell \sqrt{2\varepsilon_*} \|\delta X\|^2.$$

Absorbing constants into the $O(\cdot)$ notation gives

$$\mathcal{J}_\ell(X_\ell^* + \delta X) = \mathcal{J}_\ell(X_\ell^*) + \langle \nabla \mathcal{J}_\ell(X_\ell^*), \delta X \rangle + \frac{1}{2} \langle \delta X, K_\ell \delta X \rangle + O(\sqrt{\varepsilon_*} \|\delta X\|^2) + O(\|\delta X\|^3).$$

Thus, when the linear term is small, the local low-loss set is approximated by the ellipsoid induced by the quadratic form

$$g_\ell(\delta X, \delta X) = \langle \delta X, K_\ell \delta X \rangle.$$

If $K_\ell u_i = \lambda_i u_i$, then the quadratic approximation has radius proportional to $\lambda_i^{-1/2}$ in direction u_i, so high-curvature directions are thin and low-curvature directions are thick.

3. Prediction-null directions. If

$$\delta X \in \ker \bar{J}_\ell,$$

then

$$D\bar{\Psi}_\ell(X_\ell^*)[\delta X] = 0.$$

Thus such directions do not change the centered target logits to first order. They are therefore first-order prediction-null directions.

If $\bar{J}_\ell$ has locally constant rank r, the constant-rank theorem implies that $\ker \bar{J}_\ell$ forms a local null foliation of dimension $\dim(\mathcal{X}_t) - r$. The first-order observable variation is represented on the quotient

$$\mathcal{O}_\ell = \mathcal{X}_t / \ker \bar{J}_\ell,$$

or equivalently on any chosen r-dimensional complement of the null space. □

Also based on the Theorem 1 we have:

Corollary 1 (Teacher-KL local geometry)**.** *Let $X_\ell^\star$ be a reference hidden state and set*

$$q = p^\star := \mathrm{softmax}(\Psi_\ell(X_\ell^\star)).$$

Then $X_\ell^\star$ is a stationary point of the teacher-KL loss-to-go $J_\ell(X) = D_{\mathrm{KL}}(p^\star \| \mathrm{softmax}(\Psi_\ell(X)))$. In particular,

$$\nabla J_\ell(X_\ell^\star) = 0.$$

Consequently, under the assumptions of Theorem 1,

$$J_\ell(X_\ell^\star + \delta X) - J_\ell(X_\ell^\star) = \frac{1}{2}\langle \delta X, K_\ell \delta X \rangle + O(\sqrt{\epsilon^\star}\|\delta X\|^2) + O(\|\delta X\|^3).$$

For the usual supervised cross-entropy objective, $q = y$ is the one-hot next-token target. Then $p^\star - q$ is generally nonzero, even when the model predicts the correct token with high probability. The expansion in Theorem 1 therefore contains a potentially important linear term $\langle \nabla J_\ell(X_\ell^\star), \delta X \rangle$. In this setting K_ℓ should be interpreted as the local output-curvature operator rather than as a complete approximation to the supervised loss change. For this reason, our direct Fisher-validation experiments use the teacher-KL choice to isolate the curvature term. A systematic evaluation of the full linear-plus-quadratic approximation for supervised cross-entropy is left to future work.

A.3 Proof of Proposition 1

Proof. Let $P_s : \mathcal{X}_t \to \mathbb{R}^d$ be the projection onto the s-th token fiber. Its adjoint $P_s^\top : \mathbb{R}^d \to \mathcal{X}_t$ inserts a token-level perturbation into position s. By definition,

$$H_{\ell,s} = D_{X_{\ell,s}} \bar{\Psi}_\ell(X_\ell^*) = \bar{J}_\ell P_s^\top.$$

1. Fisher-Jacobian identity. The tokenwise score is

$$\kappa_{\ell,s} = \mathrm{tr}(P_s K_\ell P_s^\top).$$

Using $K_\ell = \bar{J}_\ell^\top F^* \bar{J}_\ell$, we get

$$P_s K_\ell P_s^\top = P_s \bar{J}_\ell^\top F^* \bar{J}_\ell P_s^\top = H_{\ell,s}^\top F^* H_{\ell,s}.$$

Therefore

$$\kappa_{\ell,s} = \mathrm{tr}(H_{\ell,s}^\top F^* H_{\ell,s}).$$

Since F^* is positive semidefinite,

$$\mathrm{tr}(H_{\ell,s}^\top F^* H_{\ell,s}) = \|F^{*1/2} H_{\ell,s}\|_F^2.$$

2. Causal support. In a decoder-only transformer with a standard causal mask, perturbations can propagate only forward along causal edges from earlier token positions to later or equal target positions. If token s is not in the downstream ancestor set of the target token t from layer ℓ to L, then no directed computational path connects $X_{\ell,s}$ to the target-position logits. Consequently,

$$D_{X_{\ell,s}} \bar{\Psi}_\ell(X_\ell^*) = 0,$$

that is,

$$H_{\ell,s} = 0.$$

The identity above then gives

$$\kappa_{\ell,s} = 0.$$

3. Pathwise coupling bound. Let $G_m^{(r\leftarrow u)}$ denote the token-block Jacobian of block m. By repeated application of the chain rule, the downstream Jacobian from (ℓ, s) to the target token at layer L can be written as a sum over directed causal paths:

$$D_{X_{\ell,s}} X_{L,t} = \sum_{\pi \in \Pi_{\ell:s\to t}} \prod_{(m,u\to r)\in\pi} G_m^{(r\leftarrow u)},$$

where the product is ordered along the path. Therefore,

$$\|D_{X_{\ell,s}} X_{L,t}\|_{\mathrm{op}} \le \sum_{\pi \in \Pi_{\ell:s\to t}} \prod_{(m,u\to r)\in\pi} \|G_m^{(r\leftarrow u)}\|_{\mathrm{op}}.$$

Since

$$H_{\ell,s} = DZ_t(X_L^*)\, D_{X_{\ell,s}} X_{L,t},$$

we have

$$\|H_{\ell,s}\|_{\mathrm{op}} \le \|DZ_t(X_L^*)\|_{\mathrm{op}} \sum_{\pi \in \Pi_{\ell:s\to t}} \prod_{(m,u\to r)\in\pi} \|G_m^{(r\leftarrow u)}\|_{\mathrm{op}}.$$

Finally,

$$\kappa_{\ell,s} = \mathrm{tr}(H_{\ell,s}^\top F^* H_{\ell,s}) \le \|F^*\|_{\mathrm{op}} \|H_{\ell,s}\|_F^2.$$

Using the same bound with the Frobenius norm, or absorbing the token hidden dimension into the constant if operator norms are used, yields

$$\kappa_{\ell,s} \le \|F^*\|_{\mathrm{op}} \|DZ_t(X_L^*)\|_{\mathrm{op}}^2 \left(\sum_{\pi \in \Pi_{\ell:s\to t}} \prod_{(m,u\to r)\in\pi} \|G_m^{(r\leftarrow u)}\|_{\mathrm{op}} \right)^2.$$

□

A.4 Proof of Proposition 2

Proof. Let

$$K_\ell = \sum_i \lambda_i u_i u_i^\top, \qquad \lambda_1 \ge \lambda_2 \ge \cdots \ge 0,$$

be an eigendecomposition of K_ℓ. Since $\Pi_\ell^{(k)}$ projects onto the span of $u_1, \ldots, u_k$, we can write

$$\delta X = \sum_i a_i u_i, \qquad \delta X_{\mathrm{tail}} = (I - \Pi_\ell^{(k)})\delta X = \sum_{i>k} a_i u_i.$$

Hence

$$\langle \delta X_{\mathrm{tail}}, K_\ell \delta X_{\mathrm{tail}} \rangle = \sum_{i>k} \lambda_i a_i^2 \le \lambda_{k+1} \sum_{i>k} a_i^2 = \lambda_{k+1} \|\delta X_{\mathrm{tail}}\|^2.$$

This proves (11).

For the reduced state

$$\widetilde{X} = X_\ell^* + \Pi_\ell^{(k)} \delta X,$$

Theorem 1 gives

$$\mathcal{J}_\ell(\widetilde{X}) = \mathcal{J}_\ell(X_\ell^*) + \left\langle \nabla \mathcal{J}_\ell(X_\ell^*), \Pi_\ell^{(k)} \delta X \right\rangle + \frac{1}{2} \left\langle \Pi_\ell^{(k)} \delta X, K_\ell \Pi_\ell^{(k)} \delta X \right\rangle + O\Big(\sqrt{\varepsilon_*} \|\Pi_\ell^{(k)} \delta X\|^2\Big) + O(\|\Pi_\ell^{(k)} \delta X\|^3).$$

Since $\Pi_\ell^{(k)}$ is an orthogonal projector,

$$\|\Pi_\ell^{(k)} \delta X\| \le \|\delta X\|,$$

so the remainder is bounded by

$$O\big(\sqrt{\varepsilon_*} \|\delta X\|^2\big) + O(\|\delta X\|^3).$$

Finally, the rank rule follows by using the spectral tail

$$\sum_{i>k} \lambda_i(K_\ell)$$

as a layerwise proxy for the total discarded output-sensitive curvature. If the tail is at most a fraction τ of the total curvature, then the discarded subspace contains only a controlled fraction of the layer's local predictive curvature. □

Table 3: Model checkpoints used in the experiments.

Model name in paper	Hugging Face checkpoint	Model reference	Params
SmolLM2-1.7B	`HuggingFaceTB/SmolLM2-1.7B`	[4]	1.7B
LLaMA-3.2-1B	`meta-llama/Llama-3.2-1B`	[9]	1B
Qwen2.5-3B	`Qwen/Qwen2.5-3B`	[27]	3B
Phi-3 Mini	`microsoft/Phi-3-mini-4k-instruct`	[1]	3.8B
Mistral-7B	`mistralai/Mistral-7B-v0.1`	[14]	7B
LLaMA-2-7B	`meta-llama/Llama-2-7b-hf`	[23]	7B
Gemma-2B	`google/gemma-2-2b`	[7]	2B
Gemma-7B	`google/gemma-7b`	[7]	7B
OLMo-3-7B	`allenai/OLMo-3-7B`	[19]	7B
Gemma-2-9B	`google/gemma-2-9b`	[7]	9B

B Experimental Protocol

Appendix B collects experimental details shared across the perturbation, rank-allocation, pruning, and distillation experiments. The goal of these settings is to keep the geometry measurements local and comparable across models: the transformer weights are fixed, calibration and evaluation examples are separated, and curvature quantities are estimated without materializing the full pullback-Fisher matrix.

Models. We evaluate open decoder-only language models ranging from approximately 1B to 9B parameters. All models are evaluated without fine-tuning. We disable key-value caching during curvature estimation so that hidden-state perturbations are propagated through the full downstream computation.

Datasets. We use held-out text from WikiText, OpenWebText, and FineWeb. For each dataset, we sample fixed-length token sequences and choose target positions among non-padding tokens. Calibration examples are used only to estimate geometric quantities or baseline saliency scores; evaluation examples are disjoint unless otherwise stated.

Target positions. For each sequence, the target position is sampled from the second half of the non-padding context. This avoids degenerate early-token targets while preserving causal ancestry over a substantial prefix. All token-level scores are computed only over causal ancestors of the target position.

Precision and attention implementation. Forward passes are run in the inference precision of the model, typically bfloat16. Curvature products, Rayleigh quotients, trace estimates, and bootstrap statistics are accumulated in float32. For experiments involving Jacobian-vector or vector-Jacobian products, we use eager/math attention rather than fused flash attention, because fused attention kernels are not always compatible with higher-order automatic differentiation.

Matrix-free curvature products. We never materialize the pullback Fisher matrix K_ℓ. Products $K_\ell v$ are computed by the sequence JVP-Fisher-VJP:

$$v \mapsto J_\ell v \mapsto F^\star J_\ell v \mapsto J_\ell^\top F^\star J_\ell v.$$

The middle multiplication by $F^\star$ is implemented as

$$F^\star u = p^\star \odot u - p^\star \langle p^\star, u \rangle.$$

C Numerical Estimation of Predictive Geometry

This appendix describes how we estimate the geometric quantities used in Section 4 without explicitly forming the pullback Fisher matrix K_ℓ. The key observation is that all required quantities can be computed using matrix-free products with K_ℓ, implemented by Jacobian-vector products (JVPs) and vector-Jacobian products (VJPs). This follows the standard automatic differentiation approach to matrix-free curvature products, related to Pearlmutter's Hessian-vector product method [20].

C.1 Matrix-free multiplication by K_ℓ

Recall that

$$K_\ell = \bar{J}_\ell^\top F^* \bar{J}_\ell, \qquad \bar{J}_\ell = D\bar{\Psi}_\ell(X_\ell^*).$$

For any vector $v \in \mathcal{X}_t$,

$$K_\ell v = \bar{J}_\ell^\top F^*(\bar{J}_\ell v).$$

Thus a product $K_\ell v$ can be computed in three steps:

$$u = \bar{J}_\ell v, \qquad w = F^* u, \qquad K_\ell v = \bar{J}_\ell^\top w.$$

The middle multiplication is cheap because

$$F^* u = p^* \odot u - p^* \langle p^*, u \rangle.$$

Therefore, no $|\mathcal{V}| \times |\mathcal{V}|$ Fisher matrix needs to be materialized.

Algorithm 1 Matrix-free multiplication by K_ℓ

Require: Hidden state X_ℓ^*, vector $v \in \mathcal{X}_t$

1: $u \leftarrow D\bar{\Psi}_\ell(X_\ell^*)[v]$ ▷ JVP
2: $w \leftarrow p^* \odot u - p^* \langle p^*, u \rangle$
3: $z \leftarrow D\bar{\Psi}_\ell(X_\ell^*)^\top[w]$ ▷ VJP
4: **return** z

C.2 PyTorch implementation with JVP and VJP

In PyTorch, the downstream logit map

$$\bar{\Psi}_\ell : \mathcal{X}_t \to \mathbb{R}^{|\mathcal{V}|}$$

can be represented as a function that takes a layer-ℓ hidden state and runs only the remaining transformer blocks, final normalization, and language model head. Matrix-free products with K_ℓ can then be implemented using `torch.func.jvp` and `torch.func.vjp`.

A schematic implementation is:

```
import torch
from torch.func import jvp, vjp

def centered_logits_from_layer(x_l):
    # Runs blocks ell,...,L-1, final norm, and lm_head.
    # Returns target-position logits with mean removed.
    logits = downstream_model_from_layer(x_l)      # shape: [vocab]
    return logits - logits.mean(dim=-1, keepdim=True)

def fisher_mv(p, u):
    # F u = diag(p)u - p(p^T u)
    return p * u - p * torch.sum(p * u, dim=-1, keepdim=True)

def K_mv(x_l_star, v):
    # JVP: u = J v
    logits_star, u = jvp(centered_logits_from_layer,
```

```
                        (x_l_star,),
                        (v,))
    p = torch.softmax(logits_star, dim=-1)

    # Fisher multiplication
    w = fisher_mv(p, u)

    # VJP: z = J^T w
    _, vjp_fn = vjp(centered_logits_from_layer, x_l_star)
    z, = vjp_fn(w)
    return z
```

In practice, one should avoid storing gradients for model parameters when only hidden-state derivatives are needed. The model weights are kept fixed, and `x_l_star` is treated as the differentiable input. Mixed precision can be used for the forward pass, but curvature estimation is often more stable in `float32` or with accumulation in `float32`.

For models with key-value caching, the downstream function should be defined carefully so that the perturbation at layer ℓ is applied to the hidden state whose effect is being studied. Caches that would bypass the perturbed hidden state must be disabled or recomputed.

C.3 Estimating traces and effective ranks with Hutchinson probes

For a symmetric matrix A, Hutchinson's estimator gives

$$\mathrm{tr}(A) = \mathbb{E}_\xi[\xi^\top A\xi],$$

where ξ has independent Rademacher entries, i.e. $\xi_i \in \{-1, +1\}$ with equal probability. With m probes,

$$\widehat{\mathrm{tr}}(A) = \frac{1}{m}\sum_{j=1}^{m} \xi_j^\top A\xi_j.$$

This estimator is unbiased for $\mathrm{tr}(A)$.

For $A = K_\ell$, each term requires one matrix-free product $K_\ell\xi_j$:

$$\xi_j^\top K_\ell\xi_j = \langle \xi_j, K_\ell\xi_j\rangle.$$

Thus

$$\widehat{\mathrm{tr}}(K_\ell) = \frac{1}{m}\sum_{j=1}^{m}\langle \xi_j, K_\ell\xi_j\rangle.$$

The participation-ratio effective rank used in the main text is

$$r_{\mathrm{eff}}(K_\ell) = \frac{\mathrm{tr}(K_\ell)^2}{\mathrm{tr}(K_\ell^2)}.$$

The denominator can also be estimated by Hutchinson:

$$\mathrm{tr}(K_\ell^2) = \mathbb{E}_\xi[\|K_\ell\xi\|^2],$$

so

$$\widehat{\mathrm{tr}}(K_\ell^2) = \frac{1}{m}\sum_{j=1}^{m}\|K_\ell\xi_j\|^2.$$

This requires one K_ℓ-product per probe if $K_\ell\xi_j$ is stored for both the trace and squared-trace estimates.

Algorithm 2 Hutchinson estimates of $\mathrm{tr}(K_\ell)$ and $\mathrm{tr}(K_\ell^2)$

Require: Matrix-free routine $v \mapsto K_\ell v$, number of probes m
$s_1 \leftarrow 0, \quad s_2 \leftarrow 0$
for $j = 1, \dots, m$ **do**
  Sample Rademacher probe ξ_j
  $y_j \leftarrow K_\ell \xi_j$
  $s_1 \leftarrow s_1 + \langle \xi_j, y_j \rangle$
  $s_2 \leftarrow s_2 + \langle y_j, y_j \rangle$
end for
$\widehat{\mathrm{tr}}(K_\ell) \leftarrow s_1/m$
$\widehat{\mathrm{tr}}(K_\ell^2) \leftarrow s_2/m$
$\widehat{r}_{\mathrm{eff}} \leftarrow \widehat{\mathrm{tr}}(K_\ell)^2 / \widehat{\mathrm{tr}}(K_\ell^2)$
return $\widehat{\mathrm{tr}}(K_\ell)$, $\widehat{\mathrm{tr}}(K_\ell^2)$, $\widehat{r}_{\mathrm{eff}}$

C.4 Practical implementation details

Choice of target distribution q. For local geometry around a trained model trajectory, we use either the one-hot next-token target or the model's own unperturbed prediction as a teacher distribution. The teacher-KL choice

$$q = p^*$$

removes the first-order term at the reference point and directly measures local sensitivity of the output distribution to hidden-state perturbations. The one-hot choice corresponds to the usual supervised cross-entropy geometry but may have a nonzero first-order term.

Batching. All estimates are averaged over calibration examples and target positions. For a batch $\mathcal{B}$, we estimate

$$\bar{K}_\ell = \frac{1}{|\mathcal{B}|} \sum_{(x,t) \in \mathcal{B}} K_\ell(x,t).$$

Matrix-free products with $\bar{K}_\ell$ are implemented by averaging per-example products.

Numerical precision. Curvature estimates can be sensitive to numerical precision. We run the model forward pass in the precision used for inference, but accumulate Hutchinson statistics, Lanczos inner products, and Rayleigh quotients in `float32`. For very small models or diagnostic experiments, `float64` can be used to validate the estimates.

Stopping criteria. For Lanczos, we use a fixed number of iterations q or stop when the change in the estimated leading eigenspace is below a tolerance. For Hutchinson trace estimation, we report the number of probes and use bootstrap confidence intervals over probes and calibration examples when reporting trace-derived quantities.

Memory. The method does not materialize K_ℓ. Its memory overhead is dominated by the downstream forward graph needed for JVP/VJP and by the storage required for Lanczos basis vectors. For large models, one may checkpoint downstream blocks or process layers independently.

D Additional Details for Local-Geometry and Token-Pruning Experiments

This section provides additional details for the experiments used to validate the predictive-geometry interpretation of the pullback Fisher operator. The goal of these experiments is not to show that a single Fisher-based heuristic dominates all possible alternatives, but rather to test two specific consequences of the theory: (i) the terminal prediction loss induces a measurable local geometry on intermediate hidden states, and (ii) the token-fiber trace of this geometry provides a useful loss-aware token saliency signal.

For a fixed trained decoder-only transformer, let $X_\ell^\star$ denote the residual-stream state at layer ℓ along the unperturbed forward pass. Let the downstream computation from layer ℓ to the target-position

logits be denoted by Ψ_ℓ. Around $X_\ell^\star$, the terminal predictive distribution induces the pullback Fisher operator

$$K_\ell = J_\ell^\top F^\star J_\ell,$$

where J_ℓ is the Jacobian of the downstream map from X_ℓ to centered target logits, and

$$F^\star = \mathrm{diag}(p^\star) - p^\star p^{\star\top}$$

is the softmax Fisher matrix at the reference output distribution $p^\star$. Intuitively, K_ℓ measures which perturbations of the hidden state are locally visible to the terminal predictive distribution.

D.1 Local Taylor Validation

The first experiment tests whether the local loss change induced by hidden-state perturbations is predicted by the Taylor expansion implied by the pullback Fisher geometry. This directly checks the central local-geometry claim: perturbations in hidden-state directions with large Fisher pullback curvature should produce larger changes in the terminal prediction.

For each model, layer ℓ, and target token position, we first run the model normally and record the reference hidden state $X_\ell^\star$. We then sample random perturbations δX_ℓ at several radii and inject $X_\ell^\star + \delta X_\ell$ at layer ℓ, while keeping the downstream transformer fixed. We compare the true change in terminal loss to local predictors computed at the reference trajectory.

For supervised cross-entropy, the local approximation includes both the first-order term and the pullback-Fisher quadratic term:

$$\Delta\mathrm{CE} \approx \langle \nabla_{X_\ell}\mathrm{CE}, \delta X_\ell \rangle + \frac{1}{2}\delta X_\ell^\top K_\ell \delta X_\ell.$$

We also evaluate two ablations:

$$\Delta\mathrm{CE}_{\mathrm{linear}} = \langle \nabla_{X_\ell}\mathrm{CE}, \delta X_\ell \rangle,$$

and

$$\Delta\mathrm{CE}_{\mathrm{quad}} = \frac{1}{2}\delta X_\ell^\top K_\ell \delta X_\ell.$$

This distinction is important because supervised one-hot CE is not generally stationary at the reference prediction, so the linear term need not vanish.

For teacher-KL, we instead compare the true change in the KL divergence from the reference predictive distribution to the Fisher quadratic predictor:

$$\Delta\mathrm{KL}(p^\star \,\|\, p_\delta) \approx \frac{1}{2}\delta X_\ell^\top K_\ell \delta X_\ell.$$

In this setting, the reference point is stationary, so the Fisher quadratic term is the leading local contribution.

For each layer and perturbation radius, we report the coefficient of determination R^2 and correlation between the true loss change and the corresponding local predictor. We aggregate results across models by first averaging within each model and radius, and then averaging across models so that each model receives equal weight. For visualization only, degenerate R^2 values are clipped to $[-1, 1]$, since very large negative R^2 values can occur when the true loss changes have near-zero variance. Outlier diagnostics are retained separately.

If the local predictive geometry is meaningful, we expect the teacher-KL quadratic approximation to explain a substantial fraction of the local KL change at perturbation radii where the signal is above numerical noise but still within the local regime. For supervised CE, we expect the linear-plus-quadratic approximation to outperform the quadratic-only approximation. We do not necessarily expect a large gain over the linear-only predictor at very small radii, because supervised CE can be first-order dominated.

Thus the expected qualitative pattern is:

$$\mathrm{CE}_{\mathrm{linear+quadratic}} \approx \mathrm{CE}_{\mathrm{linear}} \gg \mathrm{CE}_{\mathrm{quadratic-only}}$$

for small perturbations, while

$$\mathrm{TeacherKL}_{\mathrm{quadratic}}$$

should become predictive once the perturbation-induced KL signal is sufficiently measurable. This validates the interpretation of K_ℓ as a curvature or observability operator, rather than as a complete supervised-CE predictor by itself.

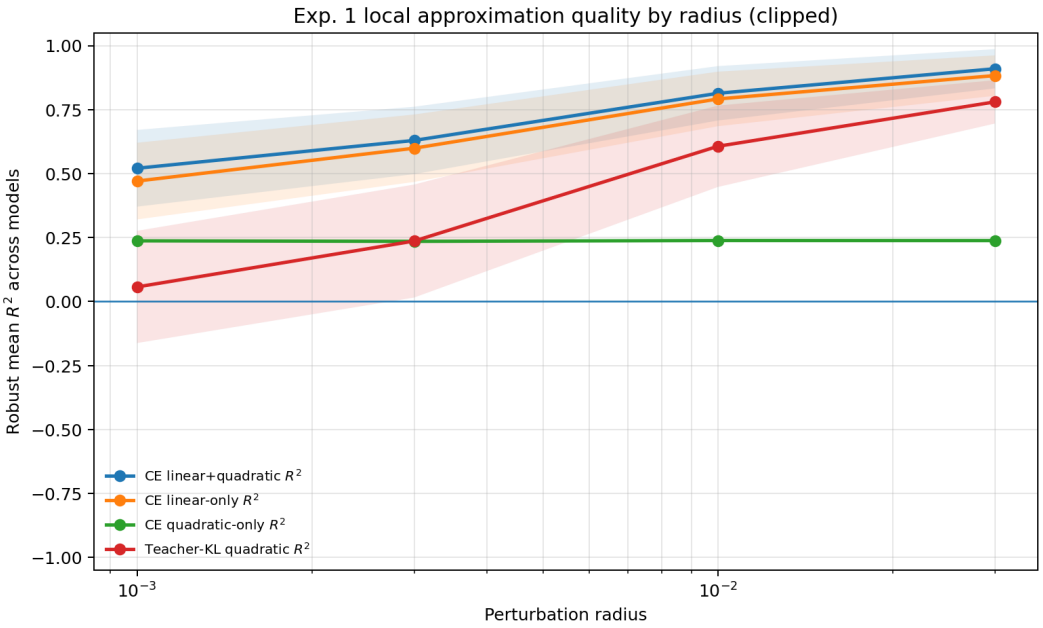

Figure 4: **Local hidden-state geometry predicts terminal loss changes.** We perturb intermediate residual-stream states at different radii and compare the true terminal loss change to local predictors induced by the downstream map. For supervised CE, the linear-only and linear-plus-quadratic predictors are nearly indistinguishable, indicating that CE is first-order dominated in this perturbation regime. The quadratic-only CE predictor is substantially weaker, as expected because supervised one-hot CE is generally not stationary at the reference prediction. In contrast, the teacher-KL quadratic predictor isolates the pullback-Fisher curvature and becomes strongly predictive at moderate perturbation radii. Curves show model-averaged R^2; for visualization, R^2 values are clipped to $[-1, 1]$ before aggregation to avoid degenerate-regression outliers.

Interpretation. A strong result for this experiment is not that the Fisher quadratic term alone predicts supervised CE. Rather, the intended conclusion is more precise: supervised CE requires the first-order term, while teacher-KL isolates the Fisher curvature. Therefore, the experiment separates two roles of the local geometry. The gradient term explains first-order supervised-loss sensitivity, whereas the pullback Fisher term captures second-order output-distribution sensitivity.

D.2 Token-Fiber Saliency and Structured Token-Update Pruning

The third experiment evaluates whether the tokenwise trace of the pullback Fisher operator provides a useful loss-aware saliency score for token updates. The theory predicts that if a token position has small pullback-Fisher trace, then perturbations localized to that token position are less visible to the terminal predictive distribution. Such token updates should therefore be safer to prune.

Let P_s denote the projection onto the residual-stream coordinates corresponding to token position s. We define the tokenwise Fisher trace

$$\kappa_{\ell,s} = \operatorname{tr}\left(P_s K_\ell P_s^\top\right).$$

Equivalently, $\kappa_{\ell,s}$ measures the squared downstream sensitivity of token s's hidden state, weighted by the output Fisher geometry. Large values indicate token positions whose hidden-state perturbations are locally visible to the final prediction, while small values indicate approximately prediction-insensitive token positions.

For each example and selected layer, we compute saliency scores over token positions. We then prune a fixed fraction ρ of token updates according to each score, selecting the least salient positions for removal. The model is evaluated after the pruning intervention, and we measure the increase in cross-entropy relative to the unpruned forward pass:

$$\Delta\mathrm{CE} = \mathrm{CE}_{\mathrm{pruned}} - \mathrm{CE}_{\mathrm{base}}.$$

Lower $\Delta\mathrm{CE}$ indicates a better pruning criterion.

In the main analysis, we report $\Delta\mathrm{CE}$ rather than $\Delta\mathrm{PPL}$. Since perplexity exponentiates CE, rare catastrophic pruning failures can dominate mean $\Delta\mathrm{PPL}$ and obscure the typical behavior. The CE difference is also equal to the log-perplexity ratio, making it a more stable primary metric:

$$\Delta\mathrm{CE} = \log \frac{\mathrm{PPL}_{\mathrm{pruned}}}{\mathrm{PPL}_{\mathrm{base}}}.$$

We compare the Fisher trace score against several alternative token saliency criteria:

- **Random**: randomly chosen token positions.
- **Activation norm**: token hidden-state norm.
- **Attention**: mean attention mass to the target position.
- $J^\top J$ **trace**: unweighted downstream Jacobian sensitivity.
- **Diagonal Fisher**: a diagonal approximation to the Fisher weighting.
- **Gradient-based scores**: first-order loss heuristics such as gradient-times-activation and top-logit gradient.

These baselines separate several possible explanations. Random pruning checks whether any signal is present at all. Activation norm tests whether large hidden activations are sufficient. Attention tests whether standard attention weights already capture the relevant token saliency. The $J^\top J$ baseline tests whether the Fisher weighting itself adds value beyond unweighted downstream sensitivity. Gradient-based scores provide strong first-order loss-aware baselines for the specific pruning intervention.

For each method and pruning fraction ρ, we report median or mean $\Delta\mathrm{CE}$ across models. Aggregation is performed by first averaging within each model, method, and pruning fraction, and then averaging across models so that each model has equal weight.

We also report a Fisher-advantage plot:

$$\Delta\mathrm{CE}_{\mathrm{baseline}} - \Delta\mathrm{CE}_{\mathrm{Fisher}}.$$

Positive values indicate that Fisher pruning causes a smaller CE increase than the baseline. This relative plot is useful for comparing Fisher directly to each alternative, but it should be interpreted together with the absolute $\Delta\mathrm{CE}$ Pareto curves.

If the tokenwise pullback-Fisher trace captures prediction-relevant information, then Fisher pruning should be substantially better than random pruning and should outperform naive activation-based criteria. We also expect it to be competitive with attention and Jacobian-style baselines. We do not require Fisher to dominate all gradient-based heuristics, because those criteria use direct first-order loss information tailored to the specific pruning intervention.

The expected qualitative ranking is therefore:

$$\mathrm{Random} \quad \text{and naive activation criteria} \quad \text{worse than Fisher},$$

while Fisher should lie near the stronger loss-aware or geometry-aware baselines:

$$\mathrm{Fisher} \approx J^\top J \approx \mathrm{Attention},$$

with direct gradient-based scores potentially competitive or stronger depending on the intervention.

Interpretation. This experiment should be interpreted as a practical validation that the pullback-Fisher trace contains usable token-level predictive information. A positive result is not necessarily that Fisher is the universally optimal pruning rule. Rather, the key conclusion is that the Fisher trace is a non-random, loss-aware geometric saliency score that identifies token updates whose removal produces smaller degradation than random or naive pruning. This supports the broader predictive-geometry view: the terminal loss induces a structured notion of local observability over token positions in the residual stream.

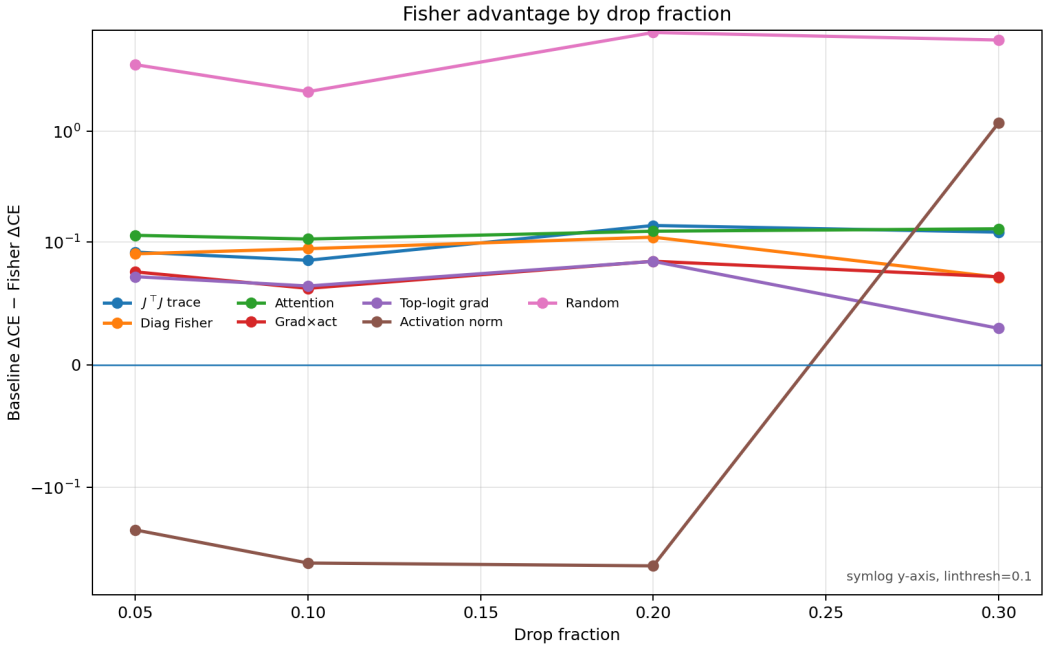

Figure 5: **Fisher token saliency is competitive with standard pruning baselines.** We prune a fraction of token updates according to each baseline score and report the Fisher advantage $\Delta\mathrm{CE}_{\mathrm{baseline}} - \Delta\mathrm{CE}_{\mathrm{Fisher}}$. Positive values indicate that Fisher pruning produces a smaller CE increase than the baseline. Fisher substantially outperforms random pruning and gives modest, consistent improvements over attention, $J^\top J$, diagonal-Fisher, and gradient-times-activation criteria in the aggregate. Activation norm is less stable across pruning fractions, performing better at small drop fractions but worse at the largest drop fraction. The y-axis uses a symlog scale with linear threshold 0.1 to show both small differences near zero and large improvements over weak baselines.

E Details for Predictive-Geometry Distillation

E.1 Low-Rank Student Construction

For each teacher model, we construct a compressed student by replacing selected linear layers with low-rank factors. We compress the attention projections

$$q_{\mathrm{proj}}, \quad k_{\mathrm{proj}}, \quad v_{\mathrm{proj}}, \quad o_{\mathrm{proj}}, \tag{18}$$

and the MLP projections

$$\mathrm{gate}_{\mathrm{proj}}, \quad \mathrm{up}_{\mathrm{proj}}, \quad \mathrm{down}_{\mathrm{proj}}. \tag{19}$$

A full linear map W is replaced by

$$W \approx AB, \qquad A \in \mathbb{R}^{d_{\mathrm{out}} \times r}, \qquad B \in \mathbb{R}^{r \times d_{\mathrm{in}}}. \tag{20}$$

Only the low-rank factors are trained during recovery; all other student parameters are frozen.

We evaluate two initializers. The first is ordinary truncated SVD. The second is activation-aware SVD. For activation-aware SVD, we collect diagonal activation second moments

$$s_i^2 = \mathbb{E}[x_i^2], \tag{21}$$

where x is the input to the corresponding linear layer. We then compute a rank-r approximation to the scaled matrix

$$W\,\mathrm{diag}(s) \tag{22}$$

and unscale the right factor. This minimizes the activation-weighted reconstruction objective

$$\|(W - \hat{W})\,\mathrm{diag}(s)\|_F^2. \tag{23}$$

This controls for the possibility that geometry-aware distillation only helps because ordinary SVD is a weak initializer.

E.2 KD Objectives

We evaluate three output-level distillation objectives.

Forward KL.

$$\mathcal{L}_{\mathrm{FKL}} = D_{\mathrm{KL}}(p_T \| p_S) = \sum_y p_T(y) \log \frac{p_T(y)}{p_S(y)}. \tag{24}$$

Reverse KL.

$$\mathcal{L}_{\mathrm{RKL}} = D_{\mathrm{KL}}(p_S \| p_T) = \sum_y p_S(y) \log \frac{p_S(y)}{p_T(y)}. \tag{25}$$

Reverse KL is a stronger autoregressive distillation baseline because it is more mode-seeking and can be better matched to a lower-capacity student.

Skew KL.

$$\mathcal{L}_{\mathrm{SkewKL}} = D_{\mathrm{KL}}\left(p_T \| \alpha p_T + (1-\alpha) p_S\right). \tag{26}$$

We use $\alpha = 0.1$ unless otherwise stated. For all KD objectives, logits are divided by temperature τ, and the KD loss is multiplied by τ^2. We use $\tau = 2$.

E.3 Predictive-Geometry Regularization

For each KD objective, we evaluate a geometry-regularized variant:

$$\mathcal{L} = \mathcal{L}_{\mathrm{KD}} + \lambda_{\mathrm{CE}} \mathcal{L}_{\mathrm{CE}} + \lambda_{\mathrm{geo}} \mathcal{L}_{\mathrm{geo}}. \tag{27}$$

The ideal geometry term is the local pullback-Fisher quadratic

$$\mathcal{L}_{\mathrm{geo}} = \frac{1}{|\mathcal{L}_{\mathrm{geo}}|} \sum_{\ell \in \mathcal{L}_{\mathrm{geo}}} \frac{1}{2} \delta h_\ell^\top K_\ell^T \delta h_\ell, \qquad \delta h_\ell = h_\ell^S - h_\ell^T, \tag{28}$$

where

$$K_\ell^T = J_\ell^\top F J_\ell \tag{29}$$

is the teacher pullback-Fisher operator induced by the downstream teacher map from layer ℓ to the output distribution.

In the current implementation, we use a readout-Fisher proxy for computational robustness across model families:

$$h_\ell \mapsto \mathrm{LMHead}(\mathrm{FinalNorm}(h_\ell)). \tag{30}$$

Thus,

$$\mathcal{L}_{\mathrm{geo}} = \frac{1}{|\mathcal{L}_{\mathrm{geo}}|} \sum_{\ell \in \mathcal{L}_{\mathrm{geo}}} \frac{1}{2} \Delta z_\ell^\top F_T \Delta z_\ell, \tag{31}$$

where

$$\Delta z_\ell = g_T(h_\ell^S) - g_T(h_\ell^T), \qquad g_T(h) = \mathrm{LMHead}_T(\mathrm{FinalNorm}_T(h)). \tag{32}$$

This preserves the output-Fisher weighting of hidden-state errors, but is weaker than the full downstream pullback-Fisher operator. We therefore report it as a readout-Fisher predictive-geometry regularizer.

E.4 Experimental Grid

We evaluate two initializers:

$$\{\mathrm{SVD}, \mathrm{ASVD}\}. \tag{33}$$

For each initializer, we evaluate:

$$\{\mathrm{FKL}, \mathrm{RKL}, \mathrm{SkewKL}, \mathrm{FKL} + \mathrm{Geo}, \mathrm{RKL} + \mathrm{Geo}, \mathrm{SkewKL} + \mathrm{Geo}\}. \tag{34}$$

The sweep is run on sub-billion-parameter models:

$$\mathrm{SmolLM2\text{-}135M}, \quad \mathrm{SmolLM2\text{-}360M}, \quad \mathrm{Qwen2.5\text{-}0.5B}. \tag{35}$$

Unless otherwise stated, we use ranks

$$r \in \{16, 32, 64\}, \tag{36}$$

sequence length 256, batch size 1, temperature $\tau = 2$, $\lambda_{\mathrm{CE}} = 0.1$, and $\lambda_{\mathrm{geo}} = 0.05$. The geometry term is evaluated every 4 optimization steps on layers

$$\{4, 8, 12, 16, 20\}. \tag{37}$$

For ASVD, activation statistics are collected from 16 calibration batches.

E.5 Metrics

We report final perplexity,

$$\mathrm{PPL}(S), \tag{38}$$

where lower is better; the compression gap to the teacher,

$$\Delta_{\mathrm{teacher}} = \mathrm{PPL}(S) - \mathrm{PPL}(T), \tag{39}$$

where lower is better; and the improvement over compressed initialization,

$$\Delta_{\mathrm{init}} = \mathrm{PPL}(S_{\mathrm{init}}) - \mathrm{PPL}(S_{\mathrm{final}}), \tag{40}$$

where higher is better. We also report the teacher-KD divergence under the corresponding KD objective, where lower is better.

E.6 Geometry Advantage

For each matched setting of model, initializer, rank, and seed, we compute

$$A_{\mathrm{geo}}(\mathcal{L}_{\mathrm{KD}}) = \mathrm{PPL}(\mathcal{L}_{\mathrm{KD}}) - \mathrm{PPL}(\mathcal{L}_{\mathrm{KD}} + \mathcal{L}_{\mathrm{geo}}). \tag{41}$$

A positive value means that adding the geometry term improves final perplexity. The main comparisons are

$$A_{\mathrm{geo}}(\mathrm{RKL}) \qquad \text{and} \qquad A_{\mathrm{geo}}(\mathrm{SkewKL}), \tag{42}$$

because they test whether geometry is complementary to stronger autoregressive KD objectives rather than only to vanilla forward-KL distillation.

E.7 Theoretical examination

We briefly justify the geometry term used in predictive-geometry distillation. Let h_ℓ^T and h_ℓ^S denote teacher and student hidden states at layer ℓ, and let

$$\delta h_\ell = h_\ell^S - h_\ell^T.$$

Let g_T be the teacher readout map used by the regularizer. In the full version g_T is the downstream teacher map from layer ℓ to the target logits; in our implementation it is the readout-Fisher proxy described above. Write

$$z_T = g_T(h_\ell^T), \qquad z_S = g_T(h_\ell^S), \qquad \Delta z = z_S - z_T,$$

and

$$p_T = \mathrm{softmax}(z_T), \qquad p_S = \mathrm{softmax}(z_S).$$

E.7.1 Exact Fisher identity in logit space.

Consider the straight path

$$z(t) = z_T + t\Delta z, \qquad p(t) = \mathrm{softmax}(z(t)).$$

Define

$$\psi(t) = D_{\mathrm{KL}}\left(p_T \| p(t)\right).$$

Then $\psi(0) = 0$, $\psi'(0) = 0$, and

$$\psi''(t) = \Delta z^\top F(p(t)) \Delta z,$$

where

$$F(p(t)) = \mathrm{diag}(p(t)) - p(t)p(t)^\top$$

is the softmax Fisher matrix. Therefore, by the integral form of Taylor's theorem,

$$D_{\mathrm{KL}}(p_T \| p_S) = \int_0^1 (1-t)\Delta z^\top F(p(t))\Delta z \, dt. \tag{43}$$

Thus the exact teacher-student KL in logit space is a path-integrated Fisher energy.

E.7.2 Local pullback to hidden space.

If g_T is differentiable at h_ℓ^T, with Jacobian

$$J_\ell = Dg_T(h_\ell^T),$$

then

$$\Delta z = J_\ell \delta h_\ell + O(\|\delta h_\ell\|^2).$$

Using (43) and smoothness of $F(p)$, we obtain

$$D_{\mathrm{KL}}(p_T \| p_S) = \frac{1}{2} \delta h_\ell^\top J_\ell^\top F(p_T) J_\ell \delta h_\ell + O(\|\delta h_\ell\|^3). \tag{44}$$

Hence the quadratic predictive-geometry term

$$\frac{1}{2} \delta h_\ell^\top K_\ell^T \delta h_\ell, \qquad K_\ell^T = J_\ell^\top F(p_T) J_\ell, \tag{45}$$

is the local hidden-space pullback of the exact teacher-student KL induced by the hidden-state mismatch. This is why the regularizer penalizes hidden errors according to their effect on the teacher predictive distribution rather than their Euclidean size.

E.7.3 When a small geometry regularizer helps.

Let

$$A(\theta) = \mathcal{L}_{\mathrm{KD}}(\theta) + \lambda_{\mathrm{CE}} \mathcal{L}_{\mathrm{CE}}(\theta)$$

be the base distillation objective, and let $G(\theta)$ be the predictive- geometry regularizer. Suppose θ_0 is a nondegenerate local minimizer of A, with Hessian

$$H_A = \nabla^2 A(\theta_0) \succ 0$$

on the trainable subspace. Let θ_λ be the nearby local minimizer of

$$A(\theta) + \lambda G(\theta)$$

for small $\lambda > 0$. By the implicit function theorem,

$$\theta_\lambda = \theta_0 - \lambda H_A^{-1} \nabla G(\theta_0) + O(\lambda^2). \tag{46}$$

Therefore, for any differentiable evaluation functional $V(\theta)$,

$$V(\theta_\lambda) - V(\theta_0) = -\lambda \left\langle \nabla V(\theta_0), H_A^{-1} \nabla G(\theta_0) \right\rangle + O(\lambda^2). \tag{47}$$

Thus, for sufficiently small λ, the geometry term improves V whenever

$$\left\langle \nabla V(\theta_0), H_A^{-1} \nabla G(\theta_0) \right\rangle > 0. \tag{48}$$

In particular, if $V = G$, then

$$G(\theta_\lambda) - G(\theta_0) = -\lambda \nabla G(\theta_0)^\top H_A^{-1} \nabla G(\theta_0) + O(\lambda^2). \tag{49}$$

Hence, whenever $\nabla G(\theta_0) \neq 0$, a sufficiently small geometry weight locally decreases the predictive-geometry mismatch. The regularizer is therefore expected to help when the compressed student has hidden-state errors with large Fisher-weighted energy that are not already removed by the base KD objective. Conversely, if the base KD objective already controls these prediction-relevant hidden directions, or if the student error lies mostly in prediction-null directions, the geometry term may provide little benefit.

Informal summary: when geometry helps distillation

The predictive-geometry regularizer is a hidden-state surrogate for teacher-student predictive mismatch. In logit space, the exact teacher-student KL is a path-integrated Fisher energy; locally, pulling this Fisher geometry back through the teacher readout gives the quadratic penalty

$$\frac{1}{2} \delta h_\ell^\top K_\ell^T \delta h_\ell.$$

Thus the regularizer does not ask the student to match all hidden-state directions equally. It mainly penalizes errors in directions that the teacher uses for prediction.

Consequently, geometry regularization is expected to help when the compressed student makes errors with large Fisher-weighted energy, i.e. errors aligned with high-curvature directions of K_ℓ^T, and when the base KD objective has not already corrected those directions. It may help little, or even hurt, when the student errors are mostly prediction-null, when the base KD loss already controls the relevant directions, or when the geometry weight is too large.

E.8 Distillation Results

Table 2 reports the paired geometry advantage

$$A_{\mathrm{geo}}(\mathcal{L}_{\mathrm{KD}}) = \mathrm{PPL}(\mathcal{L}_{\mathrm{KD}}) - \mathrm{PPL}(\mathcal{L}_{\mathrm{KD}} + \mathcal{L}_{\mathrm{geo}}),$$

computed over matched completed settings. Positive values indicate that adding the predictive-geometry term improves the compressed student at fixed model, initializer, rank, and seed. The geometry term is not uniformly beneficial for forward KL, but it improves stronger autoregressive KD objectives more reliably: RKL+Geo improves over RKL in 9/14 matched settings, and SkewKL+Geo improves over SkewKL in 10/14 matched settings. The largest average gain is obtained for SkewKL, suggesting that predictive geometry is complementary to mode-seeking and skewed KD objectives rather than simply replacing output-level distillation.

Because the sweep contains incomplete runs, we report paired statistics only on matched completed settings. A complete listing of finished and unfinished runs is provided in the supplementary tables.

Table 4: Completed KD-geometry sweep results for Qwen2.5-0.5B. Lower is better for PPL and KD divergence; higher is better for improvement over compressed initialization.

Model	Method	Rank	Compression	Init PPL ↓	Final PPL ↓	Δ_{init} ↑
Qwen2.5-0.5B	asvd+fkl	16	40.546x	2.44e+05	581.4	2.43e+05
Qwen2.5-0.5B	asvd+rkl	16	40.546x	2.44e+05	653.0	2.43e+05
Qwen2.5-0.5B	asvd+skewkl	16	40.546x	2.44e+05	551.2	2.43e+05
Qwen2.5-0.5B	svd+fkl	16	40.546x	2.15e+06	690.9	2.15e+06
Qwen2.5-0.5B	svd+fkl_geom	16	40.546x	2.15e+06	675.9	2.15e+06
Qwen2.5-0.5B	svd+rkl	16	40.546x	2.15e+06	761.5	2.15e+06
Qwen2.5-0.5B	svd+rkl_geom	16	40.546x	2.15e+06	756.7	2.15e+06
Qwen2.5-0.5B	svd+skewkl	16	40.546x	2.15e+06	690.9	2.15e+06
Qwen2.5-0.5B	svd+skewkl_geom	16	40.546x	2.15e+06	682.3	2.15e+06
Qwen2.5-0.5B	asvd+fkl	32	20.305x	1.63e+06	590.6	1.63e+06
Qwen2.5-0.5B	asvd+fkl_geom	32	20.305x	1.63e+06	665.4	1.63e+06
Qwen2.5-0.5B	asvd+rkl	32	20.305x	1.63e+06	598.1	1.63e+06
Qwen2.5-0.5B	asvd+rkl_geom	32	20.305x	1.63e+06	601.8	1.63e+06
Qwen2.5-0.5B	asvd+skewkl	32	20.305x	1.63e+06	503.3	1.63e+06
Qwen2.5-0.5B	asvd+skewkl_geom	32	20.305x	1.63e+06	495.5	1.63e+06
Qwen2.5-0.5B	svd+fkl	32	20.305x	77866	537.6	77328
Qwen2.5-0.5B	svd+fkl_geom	32	20.305x	77866	535.9	77330
Qwen2.5-0.5B	svd+rkl	32	20.305x	77866	648.9	77217
Qwen2.5-0.5B	svd+rkl_geom	32	20.305x	77866	617.1	77249
Qwen2.5-0.5B	svd+skewkl	32	20.305x	77866	556.4	77310
Qwen2.5-0.5B	svd+skewkl_geom	32	20.305x	77866	556.4	77310
Qwen2.5-0.5B	asvd+fkl	64	10.160x	6.49e+05	324.4	6.49e+05
Qwen2.5-0.5B	asvd+fkl_geom	64	10.160x	6.49e+05	362.0	6.49e+05
Qwen2.5-0.5B	asvd+rkl	64	10.160x	6.49e+05	418.2	6.49e+05
Qwen2.5-0.5B	asvd+rkl_geom	64	10.160x	6.49e+05	409.2	6.49e+05
Qwen2.5-0.5B	asvd+skewkl	64	10.160x	6.49e+05	343.2	6.49e+05
Qwen2.5-0.5B	asvd+skewkl_geom	64	10.160x	6.49e+05	357.5	6.49e+05
Qwen2.5-0.5B	svd+fkl	64	10.160x	1.53e+05	509.7	1.53e+05
Qwen2.5-0.5B	svd+fkl_geom	64	10.160x	1.53e+05	500.1	1.53e+05
Qwen2.5-0.5B	svd+rkl	64	10.160x	1.53e+05	521.0	1.53e+05
Qwen2.5-0.5B	svd+rkl_geom	64	10.160x	1.53e+05	534.2	1.53e+05
Qwen2.5-0.5B	svd+skewkl	64	10.160x	1.53e+05	462.4	1.53e+05
Qwen2.5-0.5B	svd+skewkl_geom	64	10.160x	1.53e+05	453.8	1.53e+05

Table 5: Completed KD-geometry sweep results for SmolLM2 models. Lower is better for PPL and KD divergence; higher is better for improvement over compressed initialization.

Model	Method	Rank	Compression	Init PPL ↓	Final PPL ↓	Δ_{init} ↑
SmolLM2-135M	asvd+fkl	32	10.868x	1.00e+08	713.0	1.00e+08
SmolLM2-135M	asvd+fkl_geom	32	10.868x	1.00e+08	706.3	1.00e+08
SmolLM2-135M	asvd+rkl	32	10.868x	1.00e+08	831.4	1.00e+08
SmolLM2-135M	asvd+rkl_geom	32	10.868x	1.00e+08	808.3	1.00e+08
SmolLM2-135M	asvd+skewkl	32	10.868x	1.00e+08	690.9	1.00e+08
SmolLM2-135M	asvd+skewkl_geom	32	10.868x	1.00e+08	693.1	1.00e+08
SmolLM2-135M	svd+rkl_geom	32	10.868x	4.85e+08	2171	4.85e+08
SmolLM2-135M	svd+skewkl_geom	32	10.868x	4.85e+08	1850	4.85e+08
SmolLM2-135M	asvd+fkl	64	5.434x	8.94e+07	455.2	8.94e+07
SmolLM2-135M	asvd+fkl_geom	64	5.434x	8.94e+07	430.2	8.94e+07
SmolLM2-135M	asvd+rkl	64	5.434x	8.94e+07	501.7	8.94e+07
SmolLM2-135M	asvd+rkl_geom	64	5.434x	8.94e+07	486.2	8.94e+07
SmolLM2-135M	asvd+skewkl	64	5.434x	8.94e+07	423.5	8.94e+07
SmolLM2-135M	asvd+skewkl_geom	64	5.434x	8.94e+07	416.9	8.94e+07
SmolLM2-135M	svd+fkl	64	5.434x	1.52e+08	1127	1.52e+08
SmolLM2-135M	svd+fkl_geom	64	5.434x	1.52e+08	1079	1.52e+08
SmolLM2-135M	svd+rkl	64	5.434x	1.52e+08	1378	1.52e+08
SmolLM2-135M	svd+rkl_geom	64	5.434x	1.52e+08	1339	1.52e+08
SmolLM2-135M	svd+skewkl	64	5.434x	1.52e+08	1323	1.52e+08
SmolLM2-135M	svd+skewkl_geom	64	5.434x	1.52e+08	1306	1.52e+08
SmolLM2-360M	asvd+fkl	16	36.226x	1.79e+05	1089	1.78e+05
SmolLM2-360M	asvd+fkl_geom	16	36.226x	1.79e+05	1254	1.78e+05
SmolLM2-360M	asvd+rkl	16	36.226x	1.79e+05	1124	1.78e+05
SmolLM2-360M	asvd+rkl_geom	16	36.226x	1.79e+05	1082	1.78e+05
SmolLM2-360M	asvd+skewkl	16	36.226x	1.79e+05	1219	1.78e+05
SmolLM2-360M	asvd+skewkl_geom	16	36.226x	1.79e+05	939.6	1.78e+05
SmolLM2-360M	svd+fkl	16	36.226x	8.09e+07	1850	8.09e+07
SmolLM2-360M	svd+fkl_geom	16	36.226x	8.09e+07	2078	8.09e+07
SmolLM2-360M	svd+rkl	16	36.226x	8.09e+07	2199	8.09e+07
SmolLM2-360M	svd+rkl_geom	16	36.226x	8.09e+07	2039	8.09e+07
SmolLM2-360M	svd+skewkl	16	36.226x	8.09e+07	1805	8.09e+07
SmolLM2-360M	svd+skewkl_geom	16	36.226x	8.09e+07	1637	8.09e+07
SmolLM2-360M	asvd+fkl	32	18.113x	1.46e+06	726.5	1.46e+06
SmolLM2-360M	asvd+fkl_geom	32	18.113x	1.46e+06	754.4	1.46e+06
SmolLM2-360M	asvd+rkl	32	18.113x	1.46e+06	841.9	1.46e+06
SmolLM2-360M	asvd+rkl_geom	32	18.113x	1.46e+06	927.9	1.46e+06
SmolLM2-360M	asvd+skewkl	32	18.113x	1.46e+06	747.3	1.46e+06
SmolLM2-360M	asvd+skewkl_geom	32	18.113x	1.46e+06	754.4	1.46e+06
SmolLM2-360M	svd+fkl	32	18.113x	2.01e+07	1462	2.01e+07
SmolLM2-360M	svd+fkl_geom	32	18.113x	2.01e+07	1547	2.01e+07
SmolLM2-360M	svd+rkl	32	18.113x	2.01e+07	2408	2.01e+07
SmolLM2-360M	svd+rkl_geom	32	18.113x	2.01e+07	2276	2.01e+07
SmolLM2-360M	svd+skewkl	32	18.113x	2.01e+07	1880	2.01e+07
SmolLM2-360M	svd+skewkl_geom	32	18.113x	2.01e+07	1700	2.01e+07
SmolLM2-360M	asvd+fkl	64	9.057x	78849	431.6	78418
SmolLM2-360M	asvd+fkl_geom	64	9.057x	78849	449.5	78400
SmolLM2-360M	asvd+rkl	64	9.057x	78849	516.1	78333
SmolLM2-360M	asvd+rkl_geom	64	9.057x	78849	529.2	78320
SmolLM2-360M	asvd+skewkl	64	9.057x	78849	461.0	78388
SmolLM2-360M	asvd+skewkl_geom	64	9.057x	78849	367.8	78482
SmolLM2-360M	svd+fkl	64	9.057x	3.53e+07	1152	3.53e+07
SmolLM2-360M	svd+fkl_geom	64	9.057x	3.53e+07	1159	3.53e+07
SmolLM2-360M	svd+rkl	64	9.057x	3.53e+07	1453	3.53e+07
SmolLM2-360M	svd+rkl_geom	64	9.057x	3.53e+07	1495	3.53e+07
SmolLM2-360M	svd+skewkl	64	9.057x	3.53e+07	1193	3.53e+07
SmolLM2-360M	svd+skewkl_geom	64	9.057x	3.53e+07	1106	3.53e+07

Table 6: Paired geometry comparisons for stronger KD baselines. Lower PPL is better; $\Delta_{\mathrm{Geo}} > 0$ means that adding geometry improves final PPL.

Model	Init.	Rank	RKL	RKL+Geo	$\Delta_{\mathrm{Geo}} \uparrow$	SkewKL	SkewKL+Geo	$\Delta_{\mathrm{Geo}} \uparrow$
Qwen2.5-0.5B	ASVD	32	598.1	601.8	-3.8	503.3	495.5	7.8
Qwen2.5-0.5B	ASVD	64	418.2	409.2	9.1	343.2	357.5	-14.3
Qwen2.5-0.5B	SVD	16	761.5	756.7	4.8	690.9	682.3	8.6
Qwen2.5-0.5B	SVD	32	648.9	617.1	31.8	556.4	556.4	0.0
Qwen2.5-0.5B	SVD	64	521.0	534.2	-13.2	462.4	453.8	8.6
SmolLM2-135M	ASVD	32	831.4	808.3	23.1	690.9	693.1	-2.2
SmolLM2-135M	ASVD	64	501.7	486.2	15.5	423.5	416.9	6.6
SmolLM2-135M	SVD	64	1378	1339	38.4	1323	1306	16.5
SmolLM2-360M	ASVD	16	1124	1082	41.5	1219	939.6	279.5
SmolLM2-360M	ASVD	32	841.9	927.9	-86.0	747.3	754.4	-7.1
SmolLM2-360M	ASVD	64	516.1	529.2	-13.1	461.0	367.8	93.2
SmolLM2-360M	SVD	16	2199	2039	159.5	1805	1637	167.2
SmolLM2-360M	SVD	32	2408	2276	132.2	1880	1700	179.5
SmolLM2-360M	SVD	64	1453	1495	-41.6	1193	1106	86.5

F Additional figures.

F.1 Pruning results

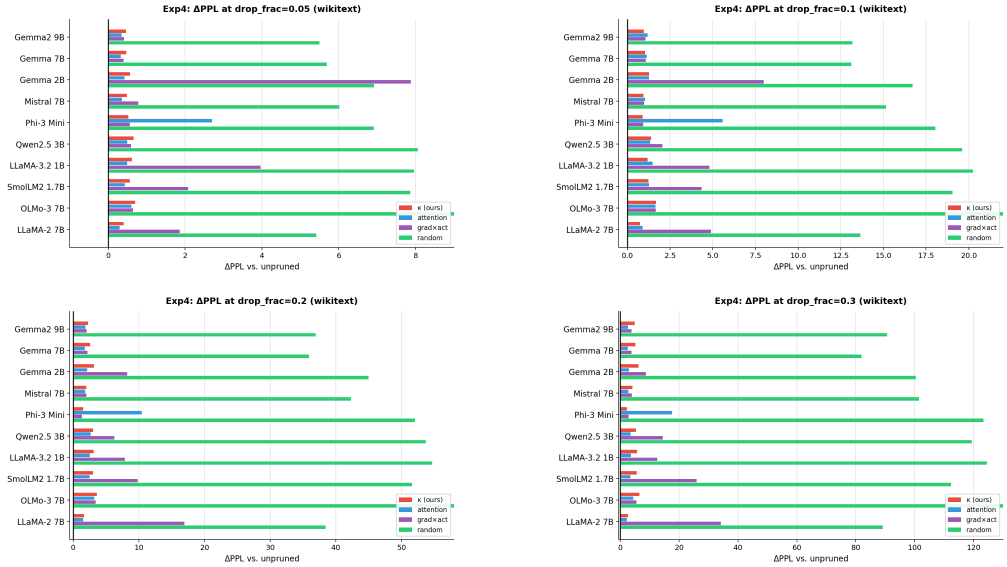

Figure 6: **Experiment 3: $\Delta\mathrm{PPL}$ on WIKITEXT under structured pruning for four drop fractions $\rho \in \{0.05, 0.10, 0.20, 0.30\}$.** We compare our geometry-guided `kappa` criterion (red) to pruning based on attention scores (blue), a gradient-activation saliency baseline (purple), and random pruning (green); lower $\Delta\mathrm{PPL}$ is better.

F.2 Direct validation of the quadratic approximation.

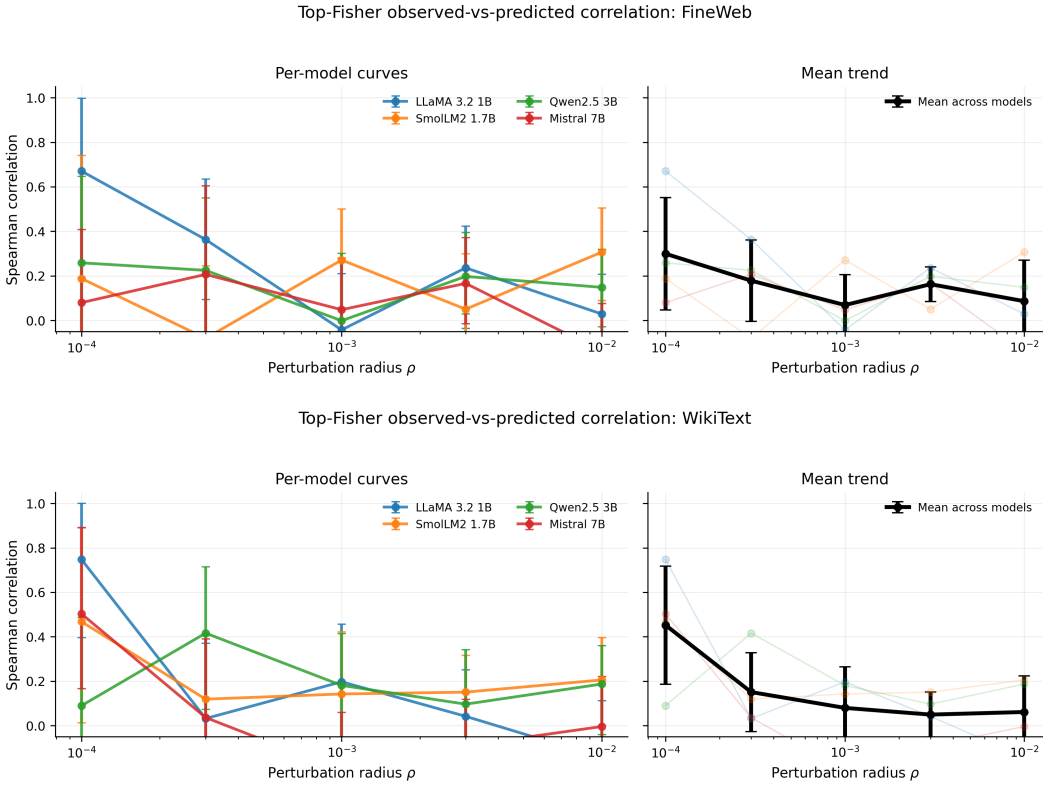

Figure 7: **Direct validation of the local pullback-Fisher approximation for top-Fisher perturbation directions on WikiText and FineWeb.** For each dataset, the left panel shows the Spearman correlation between the observed teacher-KL increase and the quadratic prediction $\frac{1}{2}\alpha^2 v^\top K_\ell v$ for each model separately, as a function of the relative perturbation radius ρ. The right panel shows the mean trend across models, with error bars denoting uncertainty across models. Correlations are highest at the smallest perturbation radii and generally decrease as ρ grows, consistent with the local nature of Theorem 1.

F.3 Compute costs

Table 7: Representative compute cost for geometry estimation on one dataset. GPU hours are approximate wall-clock estimates for computing all reported geometry statistics on the selected layer subset. Actual runtime depends on GPU type, attention implementation, and model checkpoint.

Model	Seq. len.	Examples	Model layers	GPU hours
LLaMA-3.2-1B	512	32	16	≈ 0.8
SmolLM2-1.7B	512	32	24	≈ 1.0
Gemma-2B	512	32	26	≈ 1.2
Qwen2.5-3B	512	32	36	≈ 2.0
Phi-3 Mini	512	32	32	≈ 2.2
LLaMA-2-7B	512	32	32	≈ 4.0
Mistral-7B	512	32	32	≈ 4.0
Gemma-7B	512	32	28	≈ 4.5
OLMo-3-7B	512	32	32	≈ 4.5
Gemma2-9B	512	32	42	≈ 6.0

G Additional Validation of the Local Geometry

This appendix reports the full results behind Experiment 1b (Section 4.3). All tables, figures, and per-layer results are also available at `https://github.com/timmurk/Predictive-Geometry-of-Hidden-Trajectories-in-Transformers`.

Setup. We use LLaMA-2-7B and Mistral-7B on OPENWEBTEXT, all 32 decoder layers, and three evaluation sequences per model. Curvatures are measured with centered second differences $\left[\mathcal{J}(X_\ell^\star+\alpha v)+\mathcal{J}(X_\ell^\star-\alpha v)-2\mathcal{J}(X_\ell^\star)\right]/\alpha^2$ for unit directions v, with the step α set by the relative radius ρ. We report $\rho = 10^{-3}$: smaller radii are dominated by numerical noise, while larger radii leave the local regime (cf. Figure 7).

Direction families. *top_fisher*: leading eigendirections of the single-target operator $K_{\ell,t}$. *grad_aligned*: the normalized supervised-CE gradient $\nabla\mathcal{J}_\ell(X_\ell^\star)$. *projected_null*: Gaussian directions projected onto the orthogonal complement of the leading eigenspace of $K_{\ell,t}$. *random*: isotropic Gaussian directions. *horizon_top* and *horizon_projected_null*: the analogous directions for the horizon operator $K_\ell^H = \sum_{t'\in H} K_{\ell,t'}$, where H contains the target and a window of subsequent positions. *Data-driven null directions*: the residual update written by block ℓ, the difference between the layer-ℓ hidden states of two different inputs, and the CE gradient, each projected onto the single-target null space and renormalized; the *null energy* is the fraction of the original direction's squared norm retained by this projection.

Hessian versus Fisher. Table 8 compares the full supervised-CE Hessian with the pullback Fisher operator. On top-Fisher and gradient-aligned directions the two quadratic forms have the same order of magnitude, the vectors $H_{\mathrm{CE}}v$ and $K_\ell v$ are well aligned, and $v^\top K_\ell v$ is an almost perfect rank predictor of the measured teacher-KL curvature (Spearman 0.98–0.99). Figure 8 shows that the relationship is close to the identity for small and moderate $v^\top K_\ell v$ and saturates for the most sensitive directions, where the step α leaves the quadratic regime. Alignment is weakest in the first three layers of Mistral-7B and increases towards the readout. On projected-null and random directions both quadratic forms are five to six orders of magnitude smaller than on top directions; the relative error there is close to one because the residual R_ℓ is comparable to a numerically vanishing K_ℓ term, and the finite-difference CE curvature even changes sign, indicating that it is at the noise floor.

Table 8: Direct Hessian–Fisher comparison (OPENWEBTEXT, $\rho = 10^{-3}$). Medians over directions unless stated. Rel. err. $= |v^\top H_{\mathrm{CE}}v - v^\top K_\ell v|/|v^\top H_{\mathrm{CE}}v|$; cos is the mean cosine between $H_{\mathrm{CE}}v$ and $K_\ell v$; c_{KL} and c_{CE} are finite-difference teacher-KL and CE curvatures; ρ_S is the Spearman correlation between $v^\top K_\ell v$ and c_{KL}.

Model	Direction	n	$v^\top K_\ell v$	$v^\top H_{\mathrm{CE}} v$	Rel. err.	cos	c_{KL}	c_{CE}	ρ_S
LLaMA-2-7B	top_fisher	96	0.345	0.347	0.196	0.870	0.201	0.208	0.978
	grad_aligned	96	0.133	0.127	0.349	0.762	0.108	0.083	0.993
	projected_null	192	$6.6\cdot10^{-7}$	$4.0\cdot10^{-7}$	0.965	0.178	$2.2\cdot10^{-4}$	$-5.6\cdot10^{-4}$	0.590
	random	192	$8.2\cdot10^{-7}$	$4.7\cdot10^{-7}$	0.936	0.270	$2.1\cdot10^{-4}$	$-1.0\cdot10^{-4}$	0.647
Mistral-7B	top_fisher	96	4.617	3.924	0.298	0.638	3.273	3.713	0.975
	grad_aligned	96	4.178	1.906	0.375	0.535	2.776	2.717	0.988
	projected_null	192	$1.9\cdot10^{-5}$	$8.3\cdot10^{-6}$	0.975	0.114	$1.6\cdot10^{-3}$	$1.6\cdot10^{-2}$	0.681
	random	192	$2.3\cdot10^{-5}$	$8.9\cdot10^{-6}$	0.950	0.167	$1.6\cdot10^{-3}$	$1.4\cdot10^{-2}$	0.775

Future-position leakage. Because a perturbation of X_ℓ is read by every later position, a direction that is null for the target t may still affect predictions at $t' > t$. Table 9 reports, for each direction family, the immediate curvature at t, the curvature summed over future positions, and their sum over the horizon. Top single-target directions act almost only on the target (future-to-immediate ratio 0.04–0.13), while horizon-top directions act mostly on future positions (ratio 8.7–11.9), confirming that the two operators capture different directions. Single-target null directions have future-to-immediate ratios of 3.7 and 8.0, so their small residual effect is indeed concentrated on future positions; however, its absolute size is 0.26% (LLaMA-2-7B) and 0.18% (Mistral-7B) of that of horizon-top directions and is indistinguishable from random directions (Figure 9). Single-target null directions are therefore approximately null for the whole horizon, but horizon-level analyses should use K_ℓ^H rather than $K_{\ell,t}$.

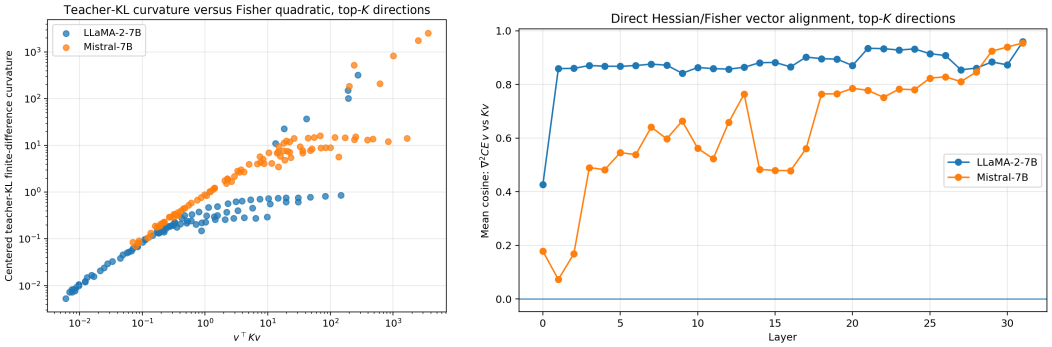

Figure 8: **Hessian–Fisher agreement on top-Fisher directions. Left:** finite-difference teacher-KL curvature versus $v^\top K_\ell v$; the relation is close to linear until the most sensitive directions leave the quadratic regime. **Right:** mean cosine between $H_{\mathrm{CE}} v$ and $K_\ell v$ by layer.

Table 9: Future-position leakage (OpenWebText, $\rho = 10^{-3}$). Medians over directions. $K_{\ell,t}$ and K_ℓ^H columns are quadratic forms; "immediate", "future" and "horizon" are finite-difference KL curvatures at the target, summed over future positions, and their total.

Model	Direction	n	$v^\top K_{\ell,t} v$	$v^\top K_\ell^H v$	Immediate	Future	Horizon	Fut./imm.
LLaMA-2-7B	single-target top	96	0.344	0.048	0.192	0.0034	0.206	0.035
	horizon top	96	0.070	0.081	0.046	0.195	0.387	8.67
	single-target null	192	$7.2 \cdot 10^{-7}$	$6.3 \cdot 10^{-7}$	$2.1 \cdot 10^{-4}$	$5.2 \cdot 10^{-4}$	$7.6 \cdot 10^{-4}$	3.68
	horizon null	192	$7.8 \cdot 10^{-7}$	$5.7 \cdot 10^{-7}$	$2.2 \cdot 10^{-4}$	$5.7 \cdot 10^{-4}$	$7.7 \cdot 10^{-4}$	3.54
	random	192	$1.1 \cdot 10^{-6}$	$7.2 \cdot 10^{-7}$	$2.0 \cdot 10^{-4}$	$5.5 \cdot 10^{-4}$	$8.0 \cdot 10^{-4}$	3.00
Mistral-7B	single-target top	96	4.775	0.868	3.034	0.477	4.683	0.131
	horizon top	96	0.881	1.508	0.546	7.340	8.299	11.88
	single-target null	192	$1.7 \cdot 10^{-5}$	$2.0 \cdot 10^{-5}$	$1.5 \cdot 10^{-3}$	$1.3 \cdot 10^{-2}$	$1.5 \cdot 10^{-2}$	7.95
	horizon null	192	$2.1 \cdot 10^{-5}$	$2.0 \cdot 10^{-5}$	$1.4 \cdot 10^{-3}$	$1.3 \cdot 10^{-2}$	$1.4 \cdot 10^{-2}$	8.36
	random	192	$2.5 \cdot 10^{-5}$	$2.0 \cdot 10^{-5}$	$1.3 \cdot 10^{-3}$	$1.3 \cdot 10^{-2}$	$1.5 \cdot 10^{-2}$	7.91

Data-driven null directions. Synthetic null directions might be unrepresentative of the perturbations that actually occur along a trajectory. Table 10 therefore projects naturally occurring directions onto the single-target null space. Block updates and inter-example differences lie almost entirely in the estimated null space (null energy ≈ 1), and their null components behave like synthetic null directions: at most 0.06% of top-direction immediate sensitivity and at most 0.26% of horizon-top future sensitivity. The CE gradient is a deliberate stress test: only 0.4% (LLaMA-2-7B) and 15% (Mistral-7B) of its energy is null, and the projected component still retains 10–18% of top-direction sensitivity. Loss-aligned directions are thus clearly separated from the null space, which shows that the estimated null space is not a sink that absorbs arbitrary directions.

Table 10: Data-driven null directions (OpenWebText, $\rho = 10^{-3}$). Medians over directions. "% top imm." is relative to the immediate curvature of single-target top directions; "% hor. fut." is relative to the future curvature of horizon-top directions.

Model	Direction	n	Immediate	Future	Null energy	% top imm.	% hor. fut.
LLaMA-2-7B	block update (null)	96	$1.6 \cdot 10^{-5}$	$4.7 \cdot 10^{-4}$	1.00	0.013	0.26
	inter-example diff. (null)	96	$1.4 \cdot 10^{-5}$	$4.7 \cdot 10^{-4}$	1.00	0.011	0.26
	random (null)	192	$1.8 \cdot 10^{-5}$	$4.8 \cdot 10^{-4}$	1.00	0.014	0.26
	CE gradient (null)	96	$1.3 \cdot 10^{-2}$	$1.3 \cdot 10^{-3}$	0.0037	9.9	0.69
Mistral-7B	block update (null)	96	$1.7 \cdot 10^{-3}$	$1.4 \cdot 10^{-2}$	1.00	0.056	0.096
	inter-example diff. (null)	96	$1.2 \cdot 10^{-3}$	$9.2 \cdot 10^{-3}$	1.00	0.040	0.063
	random (null)	192	$1.5 \cdot 10^{-3}$	$1.1 \cdot 10^{-2}$	1.00	0.048	0.072
	CE gradient (null)	96	0.549	0.552	0.153	18.0	3.78

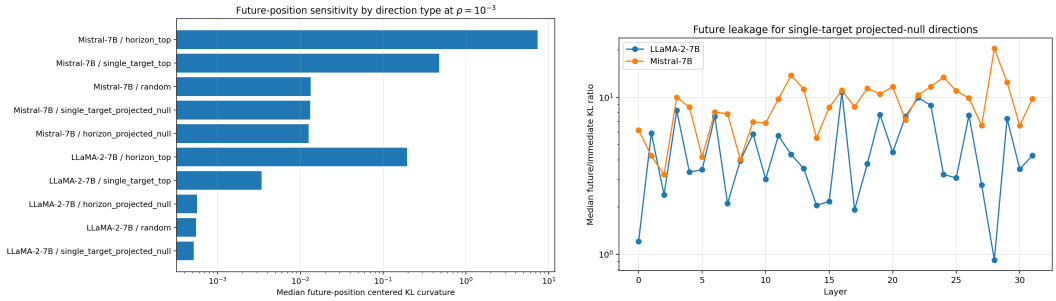

Figure 9: **Future-position sensitivity. Left:** median future-position KL curvature by direction family (log scale). **Right:** future-to-immediate ratio of single-target null directions by layer.

Independent effective-rank estimate. The participation ratio in Section 4.2 is computed from Hutchinson estimates of $\mathrm{tr}\, K_\ell$ and $\mathrm{tr}\, K_\ell^2$ (Appendix C.3). As a check that does not rely on JVP/VJP products, we also estimate r_{eff} from moments of finite-difference teacher-KL curvatures along random probes (Table 11, Figure 10). Both estimators place the effective observable rank at $O(10)$–$O(30)$ directions, i.e., at most about 1% of the hidden dimension $d = 4096$, and agree closely in mean for Mistral-7B. Their layerwise profiles, however, do not agree (Spearman ≈ -0.3): the Hutchinson estimate increases smoothly with depth, whereas the finite-difference estimate, based on three sequences and second differences, fluctuates strongly across layers. We therefore treat the low magnitude of r_{eff} as robust and the depth trend in Figure 2 as estimator-dependent.

Table 11: Effective observable rank from finite-difference moments (FD) and Hutchinson probes, over 32 layers.

Model	FD mean	FD median	Hutch. mean	Hutch. median	Med. rel. diff.	Spearman
LLaMA-2-7B	22.7	20.1	10.1	7.2	5.34	-0.30
Mistral-7B	30.3	22.9	33.2	13.3	1.22	-0.32

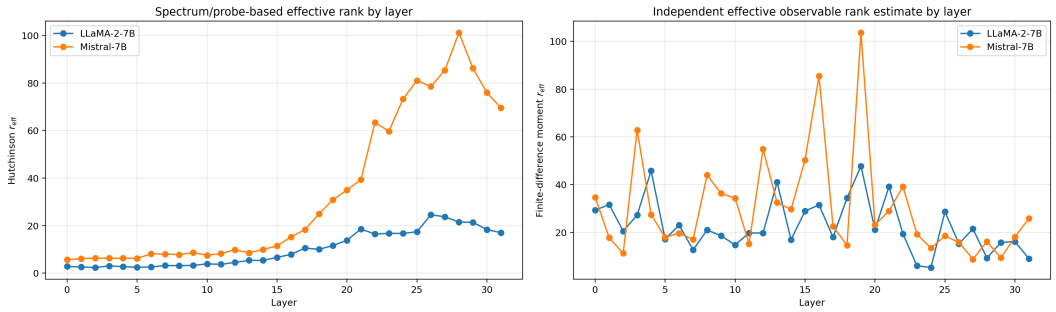

Figure 10: **Effective observable rank by layer. Left:** Hutchinson estimate. **Right:** finite-difference moment estimate. Both stay far below $d = 4096$, but only the Hutchinson estimate shows a clear depth trend.

H Limitations and Future Directions

This work focuses on a local predictive-geometric description of hidden trajectories in trained decoder-only transformers. We view the following points as natural directions for extending the framework.

Local nature of the theory. Our theoretical results characterize layerwise loss-to-go functions near a reference hidden trajectory $X_\ell^\star$. This local viewpoint is deliberate: it lets us connect the terminal prediction loss to measurable curvature and observability structure inside the residual stream. A

complementary direction is to understand how these local geometries fit together across larger regions of hidden-state space, and how such structure emerges during training.

Perturbation scale. The empirical Taylor validation depends on the perturbation radius. Very small radii can make the measured loss changes close to numerical or sampling noise, whereas very large radii can leave the local Taylor regime. This is why we report performance across radii rather than at a single scale. More adaptive choices of radius, possibly layer- or model-dependent, may give sharper local diagnostics.

Single-target versus horizon geometry. The operator K_ℓ is defined for a single target position. Because a hidden-state perturbation is read by all later positions, directions that are null for one target can still affect future predictions. Appendix G shows that this leakage is small in absolute terms for LLaMA-2-7B and Mistral-7B, but interventions that act on all positions at once are more naturally described by the horizon operator K_ℓ^H; extending the theory and the reduction guarantees to K_ℓ^H is a natural next step.

Scale of the additional validation. The Hessian–Fisher, future-leakage, data-driven-null, and independent effective-rank analyses in Appendix G use two 7B models, one corpus, and three evaluation sequences per model. Their conclusions are consistent across both models, but cross-corpus stability and larger sample sizes remain to be tested. In particular, the two effective-rank estimators agree on the magnitude of r_{eff} but not on its layerwise profile, so per-layer rank curves should be interpreted with care.

Architectural scope. We study fixed trained decoder-only pre-norm transformers with causal self-attention and a standard softmax readout. This setting already covers a large class of language models and gives a clean causal interpretation of the token-fiber score $\kappa_{\ell,s}$. Extending the same predictive-geometric view to encoder-decoder models, bidirectional transformers, mixture-of-experts models, retrieval-augmented models, and architectures with explicit memory is an interesting next step.

Computation of geometric quantities. The pullback-Fisher operator K_ℓ, its spectral summaries, and the tokenwise scores $\kappa_{\ell,s}$ can be estimated without materializing the full matrix, using matrix-free JVP/VJP products. Nevertheless, these estimates are more expensive than ordinary forward-pass statistics such as activation norms or attention weights. Improving the efficiency of trace, spectrum, and tokenwise-curvature estimation-for example through better probe sharing, sketching, or amortized estimators-would make the approach more practical for larger-scale deployment.

Reduction and pruning objectives. Our rank-allocation and token-pruning experiments are intended as controlled tests of whether predictive geometry provides useful loss-aware signals. The local theory suggests which directions or token updates are safer to discard near a reference trajectory, but it is not meant to replace full compression pipelines or task-specific pruning algorithms. In particular, direct gradient-based heuristics can be highly competitive for a fixed pruning intervention. A promising direction is to combine Fisher-geometric saliency with such first-order criteria, rather than treating them as mutually exclusive.

Geometry-aware distillation. The hidden-state geometry term is designed to complement output-level distillation by weighting student errors according to their predicted effect on the teacher's output distribution. Our results suggest that this is most useful with stronger autoregressive KD objectives such as reverse KL or skew KL. Further work could study how the geometry term interacts with longer training budgets, larger students, different low-rank parameterizations, and instruction-tuned or chat models.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the main theoretical and empirical contributions and explicitly describe the scope as local to neighborhoods of reference hidden-state trajectories. The claims are limited to fixed trained decoder-only transformers, local Fisher geometry, and the evaluated compression and pruning settings.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper includes a dedicated Limitations section discussing the locality of the theory, architectural scope, computational cost of estimating curvature quantities, and the fact that the reduction guarantees are local rather than global compression guarantees.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The assumptions for Theorem 1 and Propositions 1–2 are stated in the main text, including smoothness, low-loss, causal-mask, and local-neighborhood assumptions. Full proofs are provided in Appendix A.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper describes the model checkpoints, datasets, target-position sampling, calibration procedure, perturbation normalization, matrix-free curvature estimation, baselines, and evaluation metrics in the experimental section and Appendix B. The experiments use publicly available models and datasets, subject to their original access requirements.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The experiments use public datasets and public model checkpoints. We provide reproducibility instructions and code for matrix-free pullback-Fisher estimation, perturbation validation, rank-allocation experiments, pruning experiments, and baseline comparisons in the supplemental material or anonymized code archive.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The work evaluates fixed pretrained models and does not train new models. The paper specifies the datasets, splits, context length, calibration and evaluation sampling, target positions, perturbation radii, rank thresholds, pruning fractions, numerical precision, and baseline settings in Section 4 and Appendix B.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The main experimental results report bootstrap confidence intervals or standard errors over validation examples, perturbation directions, and randomized numerical estimates where applicable. The paper states how error bars are computed and uses nonparametric bootstrap intervals rather than assuming normality.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports representative wall-clock time, peak GPU memory, number of calibration tokens, number of curvature products, and hardware used for the main experiments in Table 7. We also describe the scaling of the matrix-free estimation cost with examples, layers, and probe directions.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work analyzes and compresses existing public language models using public text datasets and does not collect private data, involve human subjects, or deploy a system that makes decisions about individuals. We preserve anonymity in the submission materials.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses that geometry-guided compression and pruning may reduce the computational cost and energy use of language-model inference. It also notes the dual-use risk that cheaper inference can make both beneficial and harmful uses of language models more accessible.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release a new pretrained language model, image generator, or scraped dataset. It releases analysis and compression code for existing models, whose access conditions and safeguards are governed by the original model providers.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper identifies and cites the existing model checkpoints, datasets, and software libraries used in the experiments. We follow the access restrictions and license terms of the original model and dataset providers, including gated-access requirements where applicable.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The new asset introduced by the paper is code for estimating the proposed geometric quantities and reproducing the experiments. The released code includes documentation, commands, configuration options, and descriptions of the expected outputs; no new dataset or pretrained model checkpoint is introduced.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing, user studies, annotation tasks, or research with human subjects.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve human-subjects research, crowdsourcing, or collection of new data from individuals, so IRB approval or equivalent review is not applicable.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core methodology does not use an LLM as an original or non-standard research component.