RETHINKING THE EVALUATION OF TRAINING DATA ATTRIBUTION IN GENERATIVE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Concerns about the misuse of copyrighted and private data in training image generation models motivate training data attribution (TDA), which assigns each training sample a score measuring its influence on a generated query. TDA methods are commonly evaluated using the Linear Datamodeling Score (LDS), which aggregates attribution scores over sampled subsets to predict changes in model behavior after retraining. However, attribution scores exhibit a long-tailed magnitude distribution, with most scores close to zero. Finite sampling introduces noise into their estimation, and repeated estimates show substantially greater sampling variation relative to the observed score energy in low-influence regions. These fluctuations reflect noise-induced errors in measuring weak influences. Consequently, changing aggregation weights alone can alter LDS without changing the underlying attribution scores, undermining the reliability of method comparisons. For example, restricting the contribution of low-influence regions improves prediction of the full subset responses. Power transformations such as squaring can also improve prediction by suppressing the contribution of low-magnitude scores, giving methods an evaluation advantage without improving their attribution estimates. We propose Signal-to-Noise Ratio LDS (SNR-LDS), which models measurement noise as zero-mean Gaussian and estimates a signal-to-noise ratio for each attribution score. By excluding scores below a common SNR threshold, SNR-LDS limits the contribution of unreliable measurements and aims to reduce the evaluation advantage conferred by reweighting. Deletion experiments on CIFAR-2 Flow Matching models show that removing influencers identified by the method ranked highest by SNR-LDS increases mean query loss more than removing those identified by the method ranked highest by LDS. We further introduce controlled-source retrieval, which directly evaluates source rankings by recovering injected training samples from foreign concepts, complementing SNR-LDS. Code is available at here.

1 INTRODUCTION

Image generation models have made substantial progress, with Flow Matching (FM) (Lipman et al., 2022) and Diffusion Models supporting large text-to-image systems such as Stable Diffusion 3 (Esser et al., 2024). The use of large image collections raises concerns about copyrighted and private training content, reinforced by evidence that generative models can reproduce training images (Carlini et al., 2023). Training data attribution (TDA) helps investigate these concerns by estimating how training examples influence a generated image. This influence can be framed as a counterfactual question: how would the model's behavior on that image change if a training example were removed and the model retrained? Influence functions (IF) approximate this effect without retraining the model for each removal (Koh & Liang, 2017). Recent work has developed scalable attribution methods for diffusion models (Mlodozeniec et al., 2025; Lin et al., 2025). However, TDA for Flow Matching has received limited attention. To study attribution in this setting, we instantiate an IF-based method for FM, termed the Flow Matching Attribution Score (FMAS).

Evaluating attribution methods in the FM setting requires assessing how well their scores capture the effects of training data on a query. The Linear Datamodeling Score (LDS; Ilyas et al., 2022) tests whether these scores can predict the effects on models of changing the training set. For each query, it sums attribution scores over deleted samples to predict the query loss change after retraining, then computes the rank correlation between predicted and measured changes across sampled subsets.

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

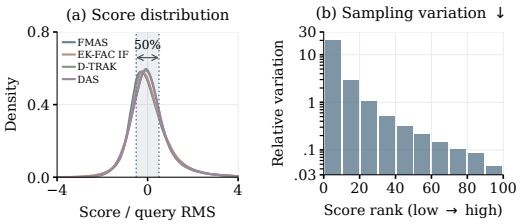


Figure 1: **Score distribution and sampling variation.** (a) CIFAR-2 FM score density; vertical lines enclose 50% of FMAS scores. (b) FMAS relative sampling variation in ten equal-sized bins, ordered from low to high absolute mean score over $R = 16$ estimates; the vertical axis is logarithmic and lower is better. Experimental setup and full results are provided in App. C.1 and C.3.

However, attribution scores across training samples exhibit a long-tailed magnitude distribution (Fig. 1(a)). Their estimation is subject to sample-independent noise arising from finite gradient sampling. Consequently, low-influence scores forming the bulk of the distribution near zero are easily perturbed by this noise, whereas large-magnitude attribution signals in the sparse tail remain identifiable. To examine perturbations in the low-influence region, we repeatedly compute attribution scores under fixed model and scoring settings and partition the training samples by their mean score magnitudes. Relative to their observed score energy, low-magnitude bins contain substantially greater sampling variation (Fig. 1(b)). Because the underlying attribution target remains fixed across repeats, these fluctuations show that noise distorts score measurements in the low-magnitude region. This makes scores near zero unreliable for aggregation into predictions of subset loss changes.

This measurement problem affects LDS through the weights assigned to scores during aggregation. Changing these weights can alter LDS while the underlying attribution scores remain fixed, weakening the reliability of method comparisons. For example, excluding the scores of low-influence samples from aggregation improves prediction of the full subset responses. Power transformations such as squaring also improve LDS by reducing the contribution of low-magnitude scores. DAS provides a concrete example: applying its terminal squaring operation to other methods eliminates its lead. Moreover, excluding the scores of low-influence samples from both the original and squared aggregations using the same rank cutoff substantially reduces the advantage of squaring. This suggests that DAS benefit largely comes from suppressing the contribution of the low-influence tail (Sec. 4.2). Full-sum LDS therefore compares methods with unequal exposure to weak-score errors, allowing score reweighting to be credited as an advantage in attribution quality.

To address this problem, we introduce *Signal-to-Noise Ratio LDS* (SNR-LDS). We model each observed attribution score as its expectation plus zero-mean Gaussian noise and estimate the noise scale from the central score distribution for each method and query. Only scores whose empirical SNR exceeds a common threshold enter subset aggregation, limiting unreliable contributions and reducing the evaluation advantage from reweighting weak scores. Our evaluation covers CIFAR Flow Matching and Diffusion Models, and Stable Diffusion 3.5 Medium fine-tuned on ArtBench-2. On CIFAR-2, deletion-and-retraining experiments show that removing influencers identified by FMAS, selected by SNR-LDS, induces greater changes in model behavior than removing those identified by DAS, selected by LDS. SNR-LDS is also robust to score reweighting. We further introduce *controlled-source retrieval*: foreign concepts from CIFAR-100 are injected into the CIFAR-10 training set while retaining the original labels, and generated images exhibiting these concepts serve as queries. These concepts form five fine-grained and five coarse-grained pairs with the original classes, allowing us to assess data attribution at different levels of semantic similarity. We evaluate existing attribution methods on SNR-LDS and controlled-source retrieval.

Our contributions are:

- We identify an attribution measurement problem in low-influence samples and show how reweighting their scores changes LDS and the comparison of attribution methods.
- We propose SNR-LDS, which uses an estimated Gaussian noise scale and a common SNR threshold to select reliable attribution scores for subsequent subset aggregation.
- We introduce controlled-source retrieval to evaluate attribution rankings through the recovery of injected training samples from the full training set.

2 Related Work

Training data attribution in generative models seeks to identify training examples that contribute to a generated output. Existing approaches include retraining-based estimation, representation matching, and post-hoc gradient-based approximation. Retraining-based methods measure model changes under modified training sets but require repeated model training. Representation matching instead compares generated and training images in a feature space, providing similarity-based evidence of their relationship. To avoid repeated retraining while accounting for model behavior, influence-function methods combine sample gradients with curvature information to approximate the parameter change caused by removing a training example and its effect on a query.

Recent work has adapted gradient-based attribution to diffusion models. Journey-TRAK (Georgiev et al., 2023) attributes generated samples at different points along denoising trajectories, examining how attribution varies across sampling steps. D-TRAK (Zheng et al., 2024) demonstrates that alternative gradient readouts can improve attribution performance over gradients of the training objective. DAS (Lin et al., 2025) derives scores from changes in model outputs and applies a terminal squaring operation. EK-FAC IF (Mlodozeniec et al., 2025) develops scalable influence approximations using structured curvature tailored to diffusion objectives. These methods differ in their attribution targets, gradient features, curvature approximations, and final score transformations. Our work examines how sampling noise and score aggregation affect their evaluation, and instantiates an influence-function-based method for Flow Matching. Appendix B.1 provides a broader discussion.

3 Preliminaries

3.1 Flow Matching

Let $\mathbb{S} = \{\mathbf{z}^{(i)}\}_{i=1}^{N}$ be a training set, where $\mathbf{z}^{(i)} = (\mathbf{x}^{(i)}, y^{(i)})$ is an image–condition pair.[1] Flow Matching (Lipman et al., 2022) learns a velocity field v_θ that continuously transforms a noise distribution into a data distribution. With $\mathbf{x}_0 \sim \mathcal{N}(0, I)$ and $t \sim \mathrm{Unif}[0, 1]$ sampled independently, the interpolation is $\mathbf{x}_t^{(i)} = (1-t)\mathbf{x}_0 + t\mathbf{x}^{(i)}$ and the target velocity is $\mathbf{x}^{(i)} - \mathbf{x}_0$. The loss for $\mathbf{z}^{(i)}$ is

$$\ell_i(\theta) = \mathbb{E}_{t,\mathbf{x}_0}\left[\left\| v_\theta(\mathbf{x}_t^{(i)}, t) - (\mathbf{x}^{(i)} - \mathbf{x}_0)\right\|_2^2\right]. \tag{1}$$

A query image is generated by sampling $\mathbf{x}(0) \sim \mathcal{N}(0, I)$ and integrating $\mathrm{d}\mathbf{x}/\mathrm{d}t = v_\theta(\mathbf{x}, t)$ to $t = 1$.

3.2 Training Data Attribution in Flow Matching

Training data attribution in flow matching estimates how training samples influence the generation of a query image $\mathbf{x}^q$. This influence can be measured through a leave-one-out (LOO) experiment: remove one training sample, retrain the model using the same procedure, and measure the change in behavior. Let θ^* and $\theta^*_{\setminus i}$ denote the parameters trained on $\mathbb{S}$ and $\mathbb{S} \setminus \{\mathbf{z}^{(i)}\}$, respectively. Using the query loss $\mathcal{J}_q(\theta) = \ell_q(\theta)$ to measure model behavior, the LOO influence is $\Delta_{qi} = \mathcal{J}_q(\theta^*_{\setminus i}) - \mathcal{J}_q(\theta^*)$. A positive influence denotes a supporting sample whose removal increases the query loss.

Computing every LOO influence requires N retraining. Influence functions (Koh & Liang, 2017) approximate deletion through a local perturbation of the training objective $\mathcal{L}_{\mathbb{S}}(\theta) = N^{-1}\sum_i \ell_i(\theta)$. Under the local stationary-branch and regularity conditions in App. A.1, the first-order score

$$\tau_{qi}^{\mathrm{IF}} = \frac{1}{N}\nabla_\theta \mathcal{J}_q(\theta^*)^\top H^{-1} \nabla_\theta \ell_i(\theta^*) \tag{2}$$

satisfies $\Delta_{qi} = \tau_{qi}^{\mathrm{IF}} + O(N^{-2})$, where $H = \nabla_\theta^2 \mathcal{L}_{\mathbb{S}}(\theta^*)$ is the training-loss Hessian matrix.

To apply influence functions to large datasets and models, we organize their computation into two layers and instantiate this framework for Flow Matching as the Flow Matching Attribution Score (FMAS). The gradient layer specifies the functions differentiated for queries and training samples.

[1] The condition is a class label or a text prompt; unconditional models use images alone. We omit conditioning inputs from the equations.

FMAS uses Monte Carlo sampling to estimate the query and training loss gradients, yielding $\widehat{g}_q$ and $\widehat{g}_i$. The curvature layer constructs and regularizes the preconditioning operator between these gradients. FMAS approximates the Hessian with a generalized Gauss–Newton (GGN) matrix $\widehat{H}$ and applies a layerwise damping matrix δ. The resulting sample-wise attribution score is

$$\widehat{\tau}_{qi}^{\mathrm{FMAS}} = \frac{1}{N}\widehat{g}_q^{\top}(\widehat{H}+\delta)^{-1}\widehat{g}_i. \tag{3}$$

App. A.2 details the FMAS construction. For query $\mathbf{x}^q$, the vectors $\widehat{\boldsymbol{\tau}}_q = (\widehat{\tau}_{qi})_{i=1}^N \in \mathbb{R}^N$ and $\boldsymbol{\Delta}_q = (\Delta_{qi})_{i=1}^N \in \mathbb{R}^N$ denote the computed attribution scores and LOO influences, respectively.

3.3 Evaluation: Linear Datamodeling Score

The Linear Datamodeling Score (LDS; Ilyas et al., 2022) evaluates the attribution methods in App. B.2 by measuring how well aggregated attribution scores predict model behavior after retraining on sampled subsets. Given a model trained on $\mathbb{S}$, we sample M subsets $\mathbb{S}_m$ by retaining each training sample independently with probability α and retrain a model with parameters θ_m^* on each subset. The measured query loss change for query in $\mathcal{Q} = \{\mathbf{x}^q\}_{q=1}^{N_q}$ is $Y_{mq} = \widehat{\mathcal{J}}_q(\theta_m^*) - \widehat{\mathcal{J}}_q(\theta^*)$, where $\widehat{\mathcal{J}}_q$ denotes the estimated query loss. Let $d_{mi} = \mathbf{1}\{\mathbf{z}^{(i)} \notin \mathbb{S}_m\}$ indicate deletion and collect these indicators into $\mathbf{d}_m = (d_{mi})_{i=1}^N$. The predicted change is the sum of scores over deletion: $P_{mq} = \mathbf{d}_m^{\top}\widehat{\boldsymbol{\tau}}_q$. LDS evaluates subset-loss prediction from aggregated scores, rather than individual LOO accuracy, by computing the per-query Spearman correlation across subsets and then averaging:

$$\mathrm{LDS}_q = \rho_S\big((P_{mq})_{m=1}^M, (Y_{mq})_{m=1}^M\big), \qquad \mathrm{LDS} = \frac{1}{N_q}\sum_{q=1}^{N_q}\mathrm{LDS}_q. \tag{4}$$

4 Method

We first examine the measurement noise in attribution scores (Sec. 4.1), then show how this noise makes LDS sensitive to score reweighting (Sec. 4.2). We address this problem with SNR-LDS (Sec. 4.3) and complement subset-loss prediction with controlled-source retrieval (Sec. 4.4).

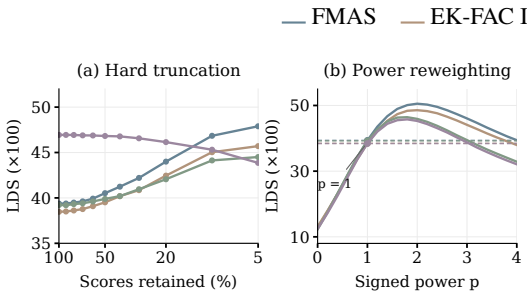
4.1 Noise in Attribution Score Estimation

TDA methods assign scores to training samples to indicate their influence on queries. We examine the score distributions of four attribution methods—FMAS, EK-FAC IF, D-TRAK, and DAS—on a CIFAR-2 FM model. After normalization, most scores lie near zero, while a small fraction have large absolute values, forming a long-tailed magnitude distribution (Fig. 1(a)). Similar patterns appear across other settings in App. C.2. The bulk of low-influence samples and the sparse high-influence tail exhibit different sensitivities to noise arising during repeated score estimation.

Finite sampling introduces randomness into gradient and Hessian estimates. For a fixed trained model, attribution method, and query q, let ω denote this randomness and $\boldsymbol{\eta}_q = \mathbb{E}_\omega[\widehat{\boldsymbol{\tau}}_q]$ the expected score vector, which we call the attribution signal. The observed scores decompose as $\widehat{\boldsymbol{\tau}}_q = \boldsymbol{\eta}_q + \boldsymbol{\varepsilon}_q$, where $\boldsymbol{\varepsilon}_q$ is measurement noise. For $\eta_{qi} \neq 0$, the relative measurement error is $|\varepsilon_{qi}|/|\eta_{qi}|$. For low-influence samples, noise comparable to $|\eta_{qi}|$ can substantially alter the measured magnitude or reverse its sign. To quantify this noise, we compute R independent score estimates under the same settings and obtain the sample mean $\bar{\tau}_{qi} = R^{-1}\sum_{r=1}^R \widehat{\tau}_{qi}^{(r)}$ and variance $s_{qi}^2 = (R-1)^{-1}\sum_{r=1}^R(\widehat{\tau}_{qi}^{(r)} - \bar{\tau}_{qi})^2$. We rank training samples by $|\bar{\tau}_{qi}|$ and partition them into ten equal-sized bins B_{qk}. The relative sampling variation within each bin is then quantified as

$$\mathrm{VarRatio}_{qk} = \frac{\sum_{i\in B_{qk}} s_{qi}^2}{\sum_{i\in B_{qk}} \bar{\tau}_{qi}^2}. \tag{5}$$

This ratio measures the strength of sampling noise relative to the observed attribution scores within each bin. A larger value indicates that sampling fluctuations are larger relative to the observed attribution signal. Figure 1(b) reports the mean $\mathrm{VarRatio}_{qk}$ across 16 FMAS generation queries,



(c) Linear and squared readouts

Readout	5%	20%	50%	100%
FMAS Lin.	47.89	44.00	40.52	39.37
FMAS Sq.	**48.65**	**50.27**	**50.96**	**51.09**
EK-FAC IF Lin.	45.70	42.45	39.51	38.47
EK-FAC IF Sq.	45.53	47.38	48.04	48.24
D-TRAK Lin.	44.50	42.05	39.89	39.22
D-TRAK Sq.	44.50	46.52	47.21	47.35
DAS Lin.	43.81	41.56	39.08	38.41
DAS Sq.	43.86	46.15	46.82	46.94

Figure 2: **Reweighting and aggregation on CIFAR-2 Flow Matching.** (a) LDS over top influencers selected by absolute score, shown from full aggregation to the top 5%. (b) Sign-preserving powers applied to pre-square scores; dashed lines mark $p = 1$. (c) Linear (Lin.) and Squared (Sq.) readouts on identical absolute-score heads. Values are LDS $\times 100$. Full results are in App. C.

with bins ordered from low to high score magnitude. The low-magnitude bins exhibit substantially greater relative sampling variation, indicating that sampling noise can obscure their attribution signals. Detailed setups and results for additional settings are reported in Appendices C.1 and C.3.

4.2 Score Reweighting and LDS

LDS aggregates scores over samples deleted from each subset, allowing noise from numerous low-influence samples to accumulate in its predictions. This aggregate noise can reverse the predicted ordering of subsets and reduce LDS (App. A.3). Reweighting low-influence samples changes their contributions to subset aggregation and can alter LDS without changing the underlying score estimates used for each query. We examine two ways to do so: excluding these samples from aggregation and reducing their relative aggregation weight through power transformations (Fig. 2).

First, we progressively exclude low-influence samples from aggregation, moving from all samples to the top 5% ranked by absolute pre-square score while retaining each method's native readout (Fig. 2(a)). This parallels TRAK's score thresholding, but selects samples by rank rather than a score cutoff (Park et al., 2023). This raises LDS for FMAS, EK-FAC IF, and D-TRAK, whereas DAS changes less and does not benefit from the same truncation, which is because DAS already squares its scores, reducing the relative weight of low-influence samples in aggregation. Motivated by this effect of squaring, we apply the sign-preserving power transformation $f_p(\widehat{\tau}_{qi}) = \mathrm{sign}(\widehat{\tau}_{qi})(|\widehat{\tau}_{qi}|/s_q)^p$, varying p to continuously adjust samples' relative contributions to aggregation. where $s_q = \|\widehat{\boldsymbol{\tau}}_q\|_\infty > 0$ normalizes scores without changing LDS at $p = 1$; DAS uses its pre-square scores. Moderate powers $p > 1$ reduce the relative contribution of low-influence samples and improve LDS, whereas $0 < p < 1$ has the opposite effect (Fig. 2(b)). Excessively large powers again reduce LDS, consistent with aggregation becoming too concentrated on a small number of samples. Additional experimental settings and detailed results appear in App. C.4.

DAS attributes its advantage to targeting model output changes, arguing that loss-based attribution introduces an additional comparison through the target (Lin et al., 2025). However, under the same conditional-mean assumption used in DAS's derivation, the residual cross term has zero expectation under the evaluation distribution, making the expected loss change equal to the expected squared output change (App. A.4). We therefore examine whether DAS's advantage instead arises primarily from squaring, which suppresses contributions from low-influence samples. We compare linear and squared aggregation for all four methods over identical, progressively smaller absolute-score heads, using DAS's pre-square scores as its linear counterpart (Fig. 2(c)). Under full aggregation, squaring improves LDS for every method and eliminates DAS's lead. As aggregation is restricted to higher-influence samples, the advantage of squaring substantially diminishes, supporting this reweighting explanation. Further comparisons across settings appear in App. C.5. These results show that LDS can reward aggregation choices even when the underlying score estimates remain unchanged, motivating a common noise-based selection criterion for comparing attribution methods.

4.3 SIGNAL-TO-NOISE RATIO LDS

To focus comparisons on samples whose influence can be measured reliably, we propose Signal-to-Noise Ratio LDS (SNR-LDS). SNR-LDS uses a criterion to exclude samples whose attribution signals are obscured by noise, reducing the effect of estimated randomness on evaluation. It models the observed score as $\widehat{\tau}_{qi} = \eta_{qi} + \varepsilon_{qi}$, with Gaussian noise $\varepsilon_{qi} \sim \mathcal{N}(0, \sigma_q^2)$. Here η_{qi} is the attribution signal defined in Sec. 4.1, and σ_q is a shared noise scale across training samples for each attribution method and query. The sample's amplitude signal-to-noise ratio is $\mathrm{SNR}_{qi} = |\eta_{qi}|/\sigma_q$. Small SNR indicates that measurement noise can obscure the sample's influence. Following the analysis above, we model the central concentration of scores near zero as noise-dominated and fit a zero-mean Gaussian there to estimate σ_q. For numerical stability, fitting uses RMS-normalized scores $x_{qi} = \widehat{\tau}_{qi}/r_q$, where $r_q = \|\widehat{\boldsymbol{\tau}}_q\|_2/\sqrt{N} > 0$; the fitted scale is then converted back to the original score units. Using the observed magnitude to estimate the signal strength, we define the empirical SNR and retain samples exceeding a common detection threshold ζ:

$$\widehat{\mathrm{SNR}}_{qi} = \frac{|\widehat{\tau}_{qi}|}{\widehat{\sigma}_q}, \qquad H_q = \{i : \widehat{\mathrm{SNR}}_{qi} > \zeta\}. \tag{6}$$

The empirical SNR is unchanged by a common positive rescaling of the scores and noise scale. We set $\zeta = 3$. The selected scores are $\widetilde{\tau}_{qi} = \widehat{\tau}_{qi}\,\mathbf{1}\{i \in H_q\}$, preserving their original values and signs while setting the remaining scores to zero before subset aggregation.

$$P_{mq}^{\mathrm{SNR}} = \sum_{i=1}^{N} d_{mi}\widetilde{\tau}_{qi}, \ \text{SNR-LDS} = \frac{1}{N_q}\sum_{q=1}^{N_q} \rho_S\big((P_{mq}^{\mathrm{SNR}})_{m=1}^{M}, (Y_{mq})_{m=1}^{M}\big)\,. \tag{7}$$

The threshold is shared across methods and queries, while the retained fraction adapts to each query's estimated noise level. This common selection rule limits the contribution of unreliable sample measurements and reduces the scope for gaining an evaluation advantage through their reweighting. More details about noise estimation and selection in SNR-LDS are specified in App. D.1.

Noise estimation and sample selection use only the attribution scores, without access to the measured subset responses. Unlike retaining a fixed fraction of top-ranked samples, SNR selection determines the aggregation range from each method and query's estimated noise scale under a common threshold. The selected set is fixed across all deletion subsets, which are evaluated against the same loss changes as in LDS. This procedure requires only one score vector per query; repeated estimates in Section 4.1 serve as sampling-noise diagnostics rather than required inputs to the evaluator.

4.4 CONTROLLED-SOURCE RETRIEVAL

We further introduce controlled-source retrieval to directly evaluate how well attribution methods rank the training sources of a generated concept. For an existing concept, we choose a foreign concept c. We replace a fraction α_{inj} of the original concept's training images with images of c and keep the original labels or text prompts. Let $\mathcal{I}_c$ be the set of replaced sample indices. These injected samples provide known retrieval targets. We train the model on this modified dataset and then select generated images that show c under the original condition as queries. For each query q, we rank all training samples by their attribution scores in the support direction, with higher scores ranked first. Let $\mathcal{R}_q(K)$ contain the indices of the top K samples. We report recall@K and average precision (AP). Recall@K is the fraction of injected samples recovered among the top K samples:

$$\mathrm{Recall}_q\,@K = \frac{|\mathcal{R}_q(K) \cap \mathcal{I}_c|}{|\mathcal{I}_c|}. \tag{8}$$

AP averages the precision at each rank occupied by an injected sample, summarizing retrieval quality across the full ranking. Let $z_{qj} = 1$ if the sample at rank j belongs to $\mathcal{I}_c$, and 0 otherwise. Then

$$\mathrm{AP}_q = \frac{1}{|\mathcal{I}_c|}\sum_{j=1}^{N} z_{qj}\left(\frac{1}{j}\sum_{h=1}^{j} z_{qh}\right). \tag{9}$$

We average both metrics over queries within each concept and then equally across concepts. Full implementation details of the Controlled-Source Retrieval evaluator are given in App. D.2.

(a) SNR-selected readouts

Readout	5%	20%	50%	100%
FMAS Lin.	44.93	44.54	44.54	44.54
FMAS Sq.	44.83	44.85	44.85	44.85
EK-FAC IF Lin.	43.46	43.31	43.31	43.31
EK-FAC IF Sq.	42.57	42.81	42.81	42.81
D-TRAK Lin.	44.50	44.50	44.50	44.50
D-TRAK Sq.	44.23	44.40	44.40	44.40
DAS Lin.	43.92	43.72	43.72	43.72
DAS Sq.	43.59	43.73	43.73	43.73

(b) Deletion effects

Deleted	Metric	Random	DAS	FMAS
300	Δ loss ($\times 100$)	0.18	0.75	1.02
	L2 distance	2.46	4.37	5.40
	CLIP Similarity	0.9595	0.8921	0.8767
1,000	Δ loss ($\times 100$)	0.25	1.09	1.94
	L2 distance	2.60	5.00	7.05
	CLIP Similarity	0.9568	0.8851	0.8477

Figure 3: **Aggregation sensitivity and deletion effects.** (a) Linear and ordinary-squared SNR-LDS readouts on CIFAR-2 Flow Matching generation queries, averaged over three model seeds. The $\zeta = 3$ SNR selection is fixed for each method and query, then intersected with progressively larger absolute-score heads; paired readouts use identical selected samples. (b) Mean deletion effects on 50 CIFAR-2 generation queries after model retraining. Results are grouped by deletion budget (300 or 1,000); CLIP similarity is the cosine similarity between image embeddings.

5 Experiments

5.1 Experimental Setup

In this section, we assess whether SNR-LDS provides more reliable comparisons of attribution methods than LDS and benchmark existing methods using SNR-LDS across Flow Matching and Diffusion Models. Controlled-source retrieval further evaluates how effectively these methods recover injected samples among their top-ranked training examples for generated queries.

We train Flow Matching models on CIFAR-2 and CIFAR-10 for attribution experiments, with a DDPM model on CIFAR-2 providing an additional evaluation setting. CIFAR-2 contains 2,500 automobile and 2,500 horse images from CIFAR-10, while CIFAR-10 uses all 50,000 training images. Stable Diffusion 3.5 Medium, a flow-matching model was also fine-tuned with LoRA on ArtBench-2, 5,000 images from the post-impressionism and ukiyo-e styles, to assess methods effectiveness on large-scale setting. For queries, each model setting uses 100 generated images and 100 validation images. Each attribution method assigns a score to every training image for each query, following App. B.2. We then construct training subsets containing approximately half of the images: 64 subsets for each CIFAR setting and 32 for ArtBench-2. Models retrained on these subsets provide the measured query loss changes. Training and subset-retraining details are given in App. E.1.

For each method and query, SNR-LDS fits a zero-mean Gaussian noise scale and aggregates scores with empirical amplitude SNR above $\zeta = 3$. The aggregated predictions are compared with the measured loss changes using Spearman correlation and averaged over queries similar to LDS. Original LDS provides a comparison using all samples to compute score on the same subsets and loss changes. Full details about the SNR-LDS implementation procedure are given in App. D.1.

5.2 Effectiveness of SNR-LDS

We first evaluate SNR-LDS's robustness to score reweighting by varying the aggregation range and comparing linear and squared readouts. For each readout, we progressively expand aggregation from the top 5% of samples ranked by absolute score to the full training set. Across all four methods, SNR-LDS with native readouts remains stable over these ranges, changing by at most 0.39 percentage points (Fig. 3(a)). Under full aggregation, replacing linear readouts with squared counterparts changes SNR-LDS by only $+0.31$, -0.50, -0.09, and $+0.01$ percentage points for FMAS, EK-FAC IF, D-TRAK, and DAS, respectively. These results show that SNR selection substantially reduces the sensitivity to aggregation range and score reweighting observed under full-sum LDS.

We then test whether the method ranked highest by SNR-LDS identifies more influential training samples than the LDS-selected method through deletion-and-retraining experiments (App. E.3). On CIFAR-2, SNR-LDS selects FMAS, whereas LDS selects DAS; both rankings use queries valid for all four methods within each model seed and weight the seed means equally. For each of 50 generated queries, we delete the top 300 or 1,000 training samples ranked by the selected method, retrain the model, and measure the change in query loss. Deleting samples ranked by FMAS yields

Method	CIFAR-2 FM		CIFAR-2 DDPM	
	Validation	Generation	Validation	Generation
Raw pixel (dot product)	NA	NA	NA	NA
Raw pixel (cosine)	NA	NA	NA	NA
CLIP similarity (dot product)	NA	NA	NA	NA
CLIP similarity (cosine)	NA	NA	NA	NA
Gradient (dot product)	1.85 ± 1.12	1.73 ± 0.64	0.04	-0.18
Gradient (cosine)	0.14 ± 0.25	0.00 ± 0.00	0.00	0.00
TracInCP	2.95 ± 2.55	0.77 ± 1.25	0.33	0.14
GAS	0.42 ± 0.21	0.00 ± 0.00	0.00	-0.05
Journey-TRAK	n/a	15.44 ± 1.67	n/a	16.33
Relative IF	32.78 ± 0.70	22.81 ± 0.22	33.71	22.76
Renormalized IF	31.34 ± 1.21	22.22 ± 1.09	26.36	16.14
TRAK	31.98 ± 1.35	21.63 ± 1.13	33.76	21.17
D-TRAK	49.05 ± 0.22	44.50 ± 1.54	39.40	30.38
DAS	50.45 ± 0.39	43.73 ± 1.45	41.19	31.02
EK-FAC IF	48.91 ± 1.70	43.31 ± 2.39	44.00	34.69
FMAS	49.60 ± 0.86	44.54 ± 1.73	44.10	34.94

Table 1: SNR-LDS (%) with $\zeta = 3$ on CIFAR-2 FM and DDPM. FM reports means and sample standard deviations across three model seeds; DDPM uses one model. Each model uses 100 queries per track, with means computed over method-specific valid queries. NA denotes no valid noise-scale fits. Valid empty selections give LDS zero. The n/a denotes its unassessed validation track. Available query coverage statistics for individual methods and query tracks are reported in Table 10.

36.49% and 77.56% larger on query-loss increases than deleting those ranked by DAS, respectively. We also regenerate images using the same sampling seeds with retrained models and compare them with the original images. Across the 50 queries, FMAS yields larger L2 distances and lower CLIP similarities at both deletion budgets (Fig. 3(b)), indicating greater changes in generated images.

Threshold sensitivity. The default amplitude threshold is $\zeta = 3$. We additionally evaluate $\zeta \in \{1, 2, 3, 4\}$ and report LDS, the retained fraction, empty selections, and method rankings at every threshold. Fixed-fraction magnitude heads provide an aggregation-range reference for this comparison. Appendix E.2 gives the complete threshold and reweighting evaluation protocol.

5.3 Attribution Benchmarking with SNR-LDS

On SNR-LDS, we benchmark post-hoc attribution methods, fixing each method's score-computation settings and applying a common SNR threshold with each query-specific noise scale. Table 1 reports results for CIFAR-2 FM and DDPM, while Table 11 covers CIFAR-10 FM and ArtBench-2. These settings span different generative model families, a larger training set, and fine-tuning of a pretrained model. Valid fits that retain no samples receive zero SNR-LDS; failed fits are reported as NA. Additional model settings and method-specific query coverage details appear in App. E.4.

On CIFAR-2 FM, FMAS and D-TRAK achieve nearly identical generation means (44.54 and 44.50), while DAS leads validation (50.45). On CIFAR-2 DDPM, FMAS slightly exceeds EK-FAC IF on both tracks, and both outperform D-TRAK and DAS. The ordering changes on CIFAR-10 FM and ArtBench-2, where DAS leads both tracks, followed by D-TRAK. The four core methods outperform TRAK and the relative and renormalized IF variants on CIFAR, but this separation is less consistent on ArtBench-2, where Journey-TRAK approaches D-TRAK on generation queries.

5.4 Controlled-Source Retrieval

Controlled-source retrieval evaluates how well attribution methods identify known training sources of a generated concept. We train CIFAR-10 models with ten foreign concepts from CIFAR-100, replacing 4% of the training images while keeping their original class labels. We construct five fine-grained pairs, such as injecting leopard images into the cat class, and five coarse-grained pairs, such

	Flow Matching		Diffusion (DDPM)	
Method	Recall@200	Global AP	Recall@200	Global AP
FMAS	0.2767	0.2531	0.2682	0.2303
EK-FAC IF	0.2607	0.2350	0.2497	0.2078
D-TRAK	0.2479	0.2349	0.3011	0.2913
DAS	0.2316	0.2116	0.2688	0.2619
TRAK	0.1182	0.0745	0.1258	0.0790
CLIP cosine	0.1429	0.1253	0.1300	0.1099
Pixel cosine	0.0221	0.0125	0.0188	0.0110
CLIP dot	0.0865	0.0680	0.0742	0.0552
Pixel dot	0.0010	0.0043	0.0009	0.0042
Gradient dot	0.0059	0.0053	0.0053	0.0059
Gradient cosine	0.0100	0.0062	0.0085	0.0065
Relative IF	0.1190	0.0726	0.1242	0.0754
Renormalized IF	0.1080	0.0681	0.0906	0.0507

Table 2: Controlled-source retrieval on CIFAR-10. Each setting uses 500 generated queries, split into 100 validation and 400 test queries, with 10 and 40 per concept. Metrics use the full 50,000-image ranking and equal concept weights. All 13 methods are evaluated on the same held-out test queries within each model setting, with injected samples serving as known retrieval targets.

as injecting sunflower images into the airplane class. Each foreign concept contributes 200 injected images. We select 500 generated queries per model using concept detection and human review, with 50 per concept. These are split into 100 validation queries for parameter selection and 400 test queries for evaluation. Each method ranks all 50,000 training images. We report recall@200 and AP, averaged over queries within each concept and then equally across concepts.

FMAS achieves the highest AP and recall@200 on FM (Table 2). EK-FAC IF has higher retrieval recall than D-TRAK and nearly identical AP. On DDPM, D-TRAK leads both retrieval metrics, followed by DAS. The recomputed clean-model SNR-LDS benchmark instead ranks DAS highest on both CIFAR-10 tracks. The evaluations use different training sets and queries: retrieval tests source recovery in the injected CIFAR-10 models, whereas SNR-LDS tests subset-loss prediction in the clean models. Their results provide complementary evidence about attribution performance.

We distinguish fine-grained injections between similar concepts (e.g., leopard into cat) from coarse-grained injections between distinct concepts (e.g., sunflower into airplane). Across both model settings, all four core methods recover sources more accurately for coarse-grained injections, suggesting that greater semantic separation facilitates source identification. On FM, FMAS leads both metrics in the coarse-grained group and D-TRAK in the fine-grained group; on DDPM, D-TRAK leads both groups. These results reveal differences across concept groups that overall rankings can obscure. App. E.5 provides the injection pairs, setup, and per-concept results.

6 Conclusion

We examined how measurement noise affects the evaluation of training data attribution in image generation models. Low-influence samples have weak attribution signals that are easily obscured by sampling noise. Restricting or reweighting their contributions to subset aggregation can improve LDS without improving the underlying influence estimates, undermining the reliability of method comparisons. We proposed SNR-LDS to select samples with reliable influence measurements through a common signal-to-noise criterion, and controlled-source retrieval to complement subset prediction with direct evaluation of source rankings. In deletion experiments under the same deletion and retraining protocol, FMAS, selected by SNR-LDS, identifies samples whose removal produces larger changes in model behavior than removing samples identified by the method selected by LDS. These findings motivate techniques that reduce sampling noise and improve the measurement of weak influences. Future work could develop more accurate attribution methods across the full training set and better account for individual influence and their joint effects in aggregation.

7 STATEMENT

AI USE STATEMENT

We used generative AI tools to assist in writing mathematical proofs. Additionally, we used these tools to edit the manuscript for readability and suggest experimental parameters. The authors reviewed all AI-assisted work, checked the mathematical arguments and their assumptions, and evaluated the suggested parameters before use. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the assistance of generative AI.

ETHICS STATEMENT

This work introduces a controlled-source retrieval dataset consisting of images generated by DDPM and Flow Matching models trained on CIFAR-10 with selected CIFAR-100 images injected into the training set. The injections retain the original class labels and establish known training sources for evaluating attribution methods. This controlled modification is intended solely for attribution research, rather than to induce harmful behavior in deployed models. The dataset construction uses existing image benchmarks and does not involve recruiting human subjects or collecting new personal information. We document the data provenance and injection procedure, and releases of data and model artifacts should respect the applicable terms of the source datasets and models.

REPRODUCIBILITY STATEMENT

We provide an anonymous GitHub link to downloadable source code covering all experiments, together with experimental artifacts including model weights, gradient features, and generated datasets. We also release the scripts and execution reports for all experiments, documenting their configurations and procedures to support reproduction of the reported results. Theoretical assumptions and derivations are provided in Appendix A, method implementations in Appendix B.2, and experimental protocols and settings in Appendices C, D, and E. Baseline methods are implemented using their publicly available code and published descriptions. We tune each method and report its best results under the documented parameter-selection protocol to support fair comparisons. The Code is available on https://anonymous.4open.science/r/SNR-LDS-65B2.

REFERENCES

Elnaz Barshan, Marc-Etienne Brunet, and Gintare Karolina Dziugaite. Relatif: Identifying explanatory training samples via relative influence. In *International Conference on Artificial Intelligence and Statistics*, pp. 1899–1909. PMLR, 2020.

Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX security symposium (USENIX Security 23)*, pp. 5253–5270, 2023.

Zheng Dai and David K Gifford. Training data attribution for diffusion models. *arXiv preprint arXiv:2306.02174*, 2023.

Bradley Efron. Microarrays, empirical bayes and the two-groups model. *Statistical Science*, 23(1): 1–22, 2008. doi: 10.1214/07-STS236. URL https://arxiv.org/abs/0808.0572.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Thomas George, César Laurent, Xavier Bouthillier, Nicolas Ballas, and Pascal Vincent. Fast approximate natural gradient descent in a kronecker factored eigenbasis. *Advances in neural information processing systems*, 31, 2018.

Kristian Georgiev, Joshua Vendrow, Hadi Salman, Sung Min Park, and Aleksander Madry. The journey, not the destination: How data guides diffusion models. *arXiv preprint arXiv:2312.06205*, 2023.

Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In *International conference on machine learning*, pp. 2242–2251. PMLR, 2019.

Zayd Hammoudeh and Daniel Lowd. Identifying a training-set attack's target using renormalized influence estimation. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1367–1381, 2022.

Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Understanding predictions with data and data with predictions. In *International Conference on Machine Learning*, pp. 9525–9587. PMLR, 2022.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pp. 1885–1894. Pmlr, 2017.

Yongchan Kwon, Eric Wu, Kevin Wu, and James Y Zou. Datainf: Efficiently estimating data influence in lora-tuned llms and diffusion models. In *International Conference on Learning Representations*, volume 2024, pp. 21921–21942, 2024.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of multivariate analysis*, 88(2):365–411, 2004.

Shuangqi Li, Hieu Le, Jingyi Xu, and Mathieu Salzmann. Learning to weight parameters for training data attribution. In *International Conference on Learning Representations*, volume 2026, pp. 70540–70570, 2026.

Jinxu Lin, Linwei Tao, Minjing Dong, and Chang Xu. Diffusion attribution score: Evaluating training data influence in diffusion model. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=kuutidLf6R.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Bruno Kacper Mlodozeniec, Runa Eschenhagen, Juhan Bae, Alexander Immer, David Krueger, and Richard E. Turner. Influence functions for scalable data attribution in diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=esYrEndGsr.

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. Trak: Attributing model behavior at scale. *arXiv preprint arXiv:2303.14186*, 2023.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. *Advances in neural information processing systems*, 33: 19920–19930, 2020.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6048–6058. IEEE, 2023.

Sheng-Yu Wang, Alexei A Efros, Jun-Yan Zhu, and Richard Zhang. Evaluating data attribution for text-to-image models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7158–7169. IEEE, 2023.

Sheng-Yu Wang, Aaron Hertzmann, Alexei Efros, Richard Zhang, and Jun-Yan Zhu. Fast data attribution for text-to-image models. *Advances in Neural Information Processing Systems*, 38: 64319–64350, 2026.

Xindi Wu, Despoina Paschalidou, Jun Gao, Antonio Torralba, Laura Leal-Taixé, Olga Russakovsky, Sanja Fidler, and Jonathan Lorraine. Motion attribution for video generation. *arXiv preprint arXiv:2601.08828*, 2026.

Yutian Zhao, Chao Du, Xiaosen Zheng, Tianyu Pang, and Min Lin. Nonparametric data attribution for diffusion models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=oP3gus0Sl3.

Xiaosen Zheng, Tianyu Pang, Chao Du, Jing Jiang, and Min Lin. Intriguing properties of data attribution on diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=vKViCoKGcB.

A Attribution Derivations and Sampling-Noise Analysis

This appendix derives FMAS in Section 3, analyzes sampling noise in subset predictions for Sections 4.1–4.2, and relates loss-based and output-based attribution.

A.1 Local Leave-One-Out Approximation

This subsection derives the local leave-one-out approximation used to define FMAS in Section 3.2.

Let $\theta \in \mathbb{R}^{P_\theta}$ denote the trainable parameters and let $\mathcal{L}_{\mathbb{S}} = N^{-1}\sum_i \ell_i$. Assume $\nabla\mathcal{L}_{\mathbb{S}}(\theta^*) = 0$ and $H = \nabla^2\mathcal{L}_{\mathbb{S}}(\theta^*)$ is positive definite. For sample i, consider

$$F_i(\theta, \epsilon) = \mathcal{L}_{\mathbb{S}}(\theta) - \epsilon\ell_i(\theta), \qquad 0 \le \epsilon \le 1/N. \tag{10}$$

Assume a stationary branch $r_i(\epsilon)$ connects $r_i(0) = \theta^*$ to the deletion endpoint. On a neighborhood of this branch, assume uniformly bounded sample gradients, sample Hessians, and query gradients, with uniformly Lipschitz sample Hessians and query gradients. The inverse of $\nabla^2_\theta F_i$ is uniformly bounded there. These bounds hold uniformly in the sample sizes considered.

Along this branch, the perturbed objective has zero parameter gradient. Differentiating this stationarity condition with respect to ϵ relates the response of the parameters to the gradient of the downweighted sample:

$$\nabla\mathcal{L}_{\mathbb{S}}(r_i(\epsilon)) - \epsilon\nabla\ell_i(r_i(\epsilon)) = 0, \tag{11}$$

$$\big[\nabla^2\mathcal{L}_{\mathbb{S}}(r_i(\epsilon)) - \epsilon\nabla^2\ell_i(r_i(\epsilon))\big]r_i'(\epsilon) = \nabla\ell_i(r_i(\epsilon)). \tag{12}$$

At $\epsilon = 0$, the perturbed Hessian reduces to H, so the initial parameter response is obtained by applying its inverse to the sample gradient. A first-order expansion along the branch at $\epsilon = 1/N$ gives the parameter change induced by deleting sample i:

$$r_i'(0) = H^{-1}\nabla\ell_i(\theta^*), \qquad r_i(1/N) - \theta^* = N^{-1}H^{-1}\nabla\ell_i(\theta^*) + O(N^{-2}). \tag{13}$$

At the endpoint, the contribution of sample i is removed from the empirical objective. The remaining objective differs from the mean loss on the reduced dataset only by a positive constant, as shown by the following rescaling:

$$F_i(\theta, 1/N) = \frac{1}{N}\sum_{j\neq i}\ell_j(\theta) = \frac{N-1}{N}\mathcal{L}_{\mathbb{S}\setminus\{\mathbf{z}^{(i)}\}}(\theta). \tag{14}$$

The two objectives have the same stationary points. If retraining returns $r_i(1/N)$, a Taylor expansion translates this local parameter change into the corresponding change in query loss:

$$\mathcal{J}_q(\theta^*_{\setminus i}) - \mathcal{J}_q(\theta^*) = N^{-1}\nabla\mathcal{J}_q(\theta^*)^\top H^{-1}\nabla\ell_i(\theta^*) + O(N^{-2}). \tag{15}$$

The remainder includes the local parameter-response approximation and the query-loss expansion. Retraining to another stationary branch introduces an additional discrepancy. Curvature approximations, damping, and parameter restrictions introduce further computational error.

A.2 Constructing FMAS from the Influence Function

This subsection details the loss normalization, gradient estimation, curvature approximation, and damping used to implement the FMAS construction in Section 3.2 for training and query images.

Loss and gradient conventions. The main text uses summed squared error; the implementation averages over the D output dimensions, so $\ell_i^{\text{avg}} = \ell_i/D$, $\mathcal{J}_q^{\text{avg}} = \mathcal{J}_q/D$, and $\Delta_{qi}^{\text{avg}} = \Delta_{qi}/D$. Equation 15 holds under either convention when gradients and curvature use the same loss scale. Damping uses the same dimension-averaged loss scale as the gradients and curvature.

For a training or query image indexed by a, use the path $\mathbf{x}_t^a = (1-t)\mathbf{x}_0 + t\mathbf{x}^a$ and target $u_a = \mathbf{x}^a - \mathbf{x}_0$ from Section 3. Let $r_a = v_{\theta^*}(\mathbf{x}_t^a, t) - u_a$ and $J_a = \partial v_\theta(\mathbf{x}_t^a, t)/\partial\theta|_{\theta^*} \in \mathbb{R}^{D\times P_\theta}$, with conditioning suppressed. Differentiating under the expectation gives

$$g_a^{\text{avg}} = \frac{2}{D}\mathbb{E}_{t,\mathbf{x}_0}[J_a^\top r_a], \qquad \widehat{g}_a = \frac{2}{Dm_a}\sum_{k=1}^{m_a}(J_a^{(k)})^\top r_a^{(k)}, \tag{16}$$

where m_a is the Monte Carlo budget. For a query, $g_q^{\text{avg}} = \nabla\mathcal{J}_q^{\text{avg}}$; for a training image, $g_i^{\text{avg}} = \nabla\ell_i^{\text{avg}}$. Sampling and parameter-subspace choices are given in Appendix B.2.

Curvature and damping. The dimension-averaged squared loss has output-space Hessian $2I/D$. The generalized Gauss–Newton construction maps this curvature into parameter space through the output Jacobian and averages over training images, times, and noise:

$$H_{\mathrm{GGN}} = \frac{2}{ND} \sum_i \mathbb{E}_{t,\mathbf{x}_0}[J_i^\top J_i]. \tag{17}$$

EK-FAC (George et al., 2018) approximates this matrix using layerwise Kronecker eigenbases and re-estimated eigenvalues. For each weight or bias block b, write its orthonormal basis as Q_b and eigenvalues as Λ_b. Weight bases combine activation and output-gradient eigenbases; bias blocks use the output-gradient basis. Separate factor and eigenvalue passes estimate these quantities. For layer l, FMAS sets $\delta_l = \varrho \, \mathrm{mean}(\Lambda_l^W)$, with $\varrho > 0$ and weight-block eigenvalues Λ_l^W; its bias block uses the same scalar. With $\widetilde{g}_{a,b} = Q_b^\top \widehat{g}_{a,b}$ and δ collecting the damping blocks,

$$\widehat{\tau}_{qi}^{\mathrm{FMAS,avg}} = \frac{1}{N} \widehat{g}_q^\top (\widehat{H} + \delta)^{-1} \widehat{g}_i = \frac{1}{N} \sum_b \sum_j \frac{\widetilde{g}_{q,b,j} \widetilde{g}_{i,b,j}}{\Lambda_{b,j} + \delta_{l(b)}}. \tag{18}$$

The stored evaluation scores omit $1/N$. This common positive factor preserves sample rankings and LDS; the curvature approximation and damping determine the estimated influence direction.

A.3 Sampling Noise and Subset Prediction Errors

This subsection supports Section 4.2 by showing how score-estimation noise enters LDS predictions. For a fixed model, attribution method, and query q, use the decomposition $\widehat{\boldsymbol{\tau}}_q = \boldsymbol{\eta}_q + \boldsymbol{\varepsilon}_q$ from Section 4.1, where $\mathbb{E}_\omega[\boldsymbol{\varepsilon}_q] = \mathbf{0}$. Let $\Sigma_q = \mathrm{Cov}_\omega(\boldsymbol{\varepsilon}_q)$ be finite, and hold the deletion subsets and measured responses fixed. The prediction for subset m is $P_{mq} = \mathbf{d}_m^\top \boldsymbol{\eta}_q + \mathbf{d}_m^\top \boldsymbol{\varepsilon}_q$, so aggregation includes noise together with the attribution signal from the same set of deleted samples.

For two subsets m and n, write $\mathbf{w}_{mn} = \mathbf{d}_m - \mathbf{d}_n$. Samples deleted from both subsets cancel in this difference; only samples with different deletion indicators contribute to the predicted gap. The identities separate the signal from sampling fluctuations and give the variance:

$$\begin{aligned} P_{mq} - P_{nq} &= \mathbf{w}_{mn}^\top \boldsymbol{\eta}_q + \mathbf{w}_{mn}^\top \boldsymbol{\varepsilon}_q, \\ \mathrm{Var}_\omega(P_{mq} - P_{nq}) &= \mathbf{w}_{mn}^\top \Sigma_q \mathbf{w}_{mn}. \end{aligned} \tag{19}$$

The variance identity follows directly from the covariance of a linear combination. If score-estimation errors are uncorrelated across samples, it reduces to $\sum_i w_{mn,i}^2 (\Sigma_q)_{ii}$. Thus, fluctuations from numerous low-influence samples can accumulate even though each error has zero mean. The general expression also captures correlations induced by shared query gradients.

Suppose the mean-score prediction orders a pair correctly: $Y_{mq} > Y_{nq}$ and $\mathbf{w}_{mn}^\top \boldsymbol{\eta}_q > 0$. The observed prediction reverses this order whenever $\mathbf{w}_{mn}^\top \boldsymbol{\varepsilon}_q < -\mathbf{w}_{mn}^\top \boldsymbol{\eta}_q$. Noise can therefore disrupt subset rankings and reduce LDS. This motivates the fixed-score experiments in Section 4.2, which test whether downweighting low-influence samples improves prediction.

A.4 Loss Change and Squared Output Change

This subsection relates loss change to squared output change for the analysis in Section 4.2, making explicit the condition under which their expectations coincide.

For a fixed query, let $X_q = (\mathbf{x}_t^q, t)$ include any conditioning, let $U_q = \mathbf{x}^q - \mathbf{x}_0$, and take expectations over the same time and noise distribution for both models. Write $r_q = v_{\theta^*}(X_q) - U_q$ and $\delta v = v_{\theta'}(X_q) - v_{\theta^*}(X_q)$. Assuming finite second moments, expansion of the squared loss gives

$$\mathcal{J}_q^{\mathrm{avg}}(\theta') - \mathcal{J}_q^{\mathrm{avg}}(\theta^*) = \frac{2}{D} \mathbb{E}\langle r_q, \delta v \rangle + \frac{1}{D} \mathbb{E}\|\delta v\|_2^2. \tag{20}$$

Taking $\theta' = \theta^*_{\setminus i}$ gives the LOO loss change. It differs from squared output displacement by the residual cross term, which can have either sign and need not vanish for an individual query.

Proposition 1 (Conditional-mean predictor)**.** *For a common evaluation distribution P of (X, U), let $v(X)$, $w(X)$, and U be square-integrable and set $\mathcal{R}_P(f) = D^{-1}\mathbb{E}_P\|f(X) - U\|_2^2$. If $v(X) = \mathbb{E}_P[U \mid X]$ almost surely, the residual is orthogonal in expectation to $w - v$, and the risks satisfy*

$$\mathbb{E}_P\langle v - U, w - v \rangle = 0, \qquad \mathcal{R}_P(w) - \mathcal{R}_P(v) = \frac{1}{D} \mathbb{E}_P\|w - v\|_2^2. \tag{21}$$

Proof. Conditioning on X gives $\mathbb{E}_P\langle v-U, w-v\rangle = \mathbb{E}_P\langle v-\mathbb{E}_P[U \mid X], w-v\rangle = 0$. Substitution into the squared-loss expansion proves the identity under the stated common evaluation distribution. □

The condition applies to the evaluation distribution in the identity. A conditional-mean predictor under the training distribution need not satisfy it for a fixed query. Likewise, parameter stationarity constrains expected residual inner products with parameter derivatives on the training distribution; it does not imply orthogonality to every finite retraining displacement on the query distribution. Thus, equality of these attribution targets requires the residual cross term to vanish.

Even if a subset's output displacement were the sum of individual displacements δv_i, squaring that sum would introduce interactions between the selected samples. Expanding the squared norm makes these pairwise terms explicit without imposing independence or orthogonality assumptions:

$$\left\| \sum_i d_i \delta v_i \right\|_2^2 = \sum_i d_i \|\delta v_i\|_2^2 + 2 \sum_{i<j} d_i d_j \langle \delta v_i, \delta v_j \rangle. \tag{22}$$

Individual squared displacements therefore need not sum to the squared displacement of a subset. Their sum omits the pairwise inner products, which can increase or decrease the total depending on the alignment of the output changes associated with the selected training samples.

B Related Work and Baseline Constructions

B.1 Related Work

Retraining and gradient-based attribution. Data Shapley estimates contributions across training coalitions (Ghorbani & Zou, 2019), while datamodels predict model behavior from training-subset membership (Ilyas et al., 2022). For diffusion models, encoded ensembles enable data ablation through models trained on designed data splits (Dai & Gifford, 2023). To avoid repeated retraining, influence functions combine training and query gradients with inverse curvature (Koh & Liang, 2017), TracIn accumulates gradient agreement along training (Pruthi et al., 2020), and TRAK uses projected gradient features (Park et al., 2023). GAS and RelatIF further modify gradient or influence normalization (Hammoudeh & Lowd, 2022; Barshan et al., 2020).

Attribution in diffusion and Flow Matching models. Journey-TRAK studies attribution along denoising trajectories (Georgiev et al., 2023), D-TRAK explores alternative gradient readouts (Zheng et al., 2024), and DAS derives scores from output changes (Lin et al., 2025). Scalable influence functions use structured generalized Gauss–Newton approximations, including EK-FAC (George et al., 2018; Mlodozeniec et al., 2025), while DataInf provides an efficient influence approximation for LoRA-tuned models (Kwon et al., 2024). Parameter-weighted D-TRAK learns parameter importance within score construction (Li et al., 2026), unlike our reweighting of already computed sample scores. FMAS instantiates individual-example influence scores for FM; our evaluation examines the effects of measurement noise and score aggregation.

Representation matching and source retrieval. Representation-based approaches retrieve training sources through similarities between generated and training samples. Wang et al. (2023) use model customization toward exemplar images to construct attribution examples and evaluate feature spaces such as DINO and CLIP. FastGDA learns an efficient retrieval representation from rankings supplied by the unlearning-based AbU+ procedure (Wang et al., 2026). NDA derives patch-level similarities from an optimal diffusion score function, without model gradients or retraining (Zhao et al., 2026). Our controlled-source retrieval introduces foreign concepts while retaining the original labels and evaluates recovery of the injected sources within the complete training set.

Evaluation objectives. LDS evaluates aggregated scores through subset-retraining predictions (Park et al., 2023), whereas deletion experiments measure the effects of removing selected samples (Zheng et al., 2024; Mlodozeniec et al., 2025). These evaluations address different questions from known-source retrieval. Our analysis shows that reweighting weak scores can improve LDS without improving the underlying influence estimates. SNR-LDS addresses this sensitivity

through a common noise-based selection rule, while controlled-source retrieval evaluates source rankings without aggregating individual attribution scores over deleted training samples.

B.2 Baseline Constructions

This subsection specifies the attribution methods compared with FMAS in the experimental setup of Section 5.1, together with the gradient, curvature, and score constructions introduced in Section 3.2.

We describe the gradient, curvature, and score layers used in Section 3, followed by the additional benchmark methods. Let o_θ denote the D-dimensional velocity output for FM or noise prediction for DDPM. Sampling budgets and dataset-specific settings appear in Appendices C.1 and E.1.

B.2.1 Gradient Features

FMAS and EK-FAC IF differentiate the dimension-averaged squared loss on both the training and query sides, as in Equation 16. The parameter subspace includes convolutional and linear weights and biases, excluding GroupNorm parameters and class embeddings. Query gradients use evaluation mode; training gradients use training mode with independently sampled horizontal flips. FMAS uses stratified times and antithetic noise; EK-FAC IF samples both independently.

D-TRAK (Zheng et al., 2024) and DAS (Lin et al., 2025) differentiate scalar functions of the model output rather than the squared training loss. The readouts below specify these functions separately for D-TRAK and the two model-specific DAS implementations:

$$f_{\text{D-TRAK}}(o) = D^{-1}\|o\|_2^2, \qquad f_{\text{DAS,FM}}(o) = D^{-1}\mathbf{1}^\top o, \qquad f_{\text{DAS,DDPM}}(o) = \|o\|_1. \tag{23}$$

For image a, let $h_a^{(k)} = \nabla_\theta f(o_{\theta,a}^{(k)})|_{\theta^*}$ at the kth sampled time and noise realization. Features normalize each gradient before averaging and applying a shared random projection:

$$\bar{h}_a = \frac{1}{T}\sum_{k=1}^{T} \frac{h_a^{(k)}}{\|h_a^{(k)}\|_2 + 10^{-8}}, \qquad \phi_a = R_{\text{proj}}\bar{h}_a, \qquad R_{\text{proj}} \in \mathbb{R}^{p\times P_{\text{all}}}. \tag{24}$$

These features include all P_{all} trainable parameters in evaluation mode. Training and query features use a common readout and random projection so that both are represented in the same feature space.

B.2.2 Curvature and Native Scores

FMAS and EK-FAC IF share the EK-FAC GGN construction in Appendix A.2. Writing $\widetilde{g}_{a,b} = Q_b^\top \widehat{g}_{a,b}$, their stored bilinear score is

$$s_{qi} = \sum_b \sum_j \frac{\widetilde{g}_{q,b,j}\widetilde{g}_{i,b,j}}{\Lambda_{b,j} + d_b}. \tag{25}$$

EK-FAC IF uses a common $d_b = \lambda_{\text{global}}$; FMAS uses the layerwise damping defined above. These methods differ in gradient sampling and damping while sharing the curvature construction. Their stored scores omit the common $1/N$ factor in the influence formula.

For projected features, stack the training vectors as $G = [\phi_1^\top; \ldots; \phi_N^\top] \in \mathbb{R}^{N\times p}$. The regularized feature Gram matrix defines a shared preconditioner. After normalizing its inverse by a common positive scale, the resulting bilinear form combines the query feature with each training feature:

$$K_\lambda = G^\top G + \lambda I_p, \qquad A_\lambda = \frac{K_\lambda^{-1}}{\text{mean}\,|K_\lambda^{-1}|}, \qquad s_{qi} = \phi_q^\top A_\lambda \phi_i. \tag{26}$$

The normalization of K_λ^{-1} is a common positive scale and leaves LDS unchanged. FMAS, EK-FAC IF, and D-TRAK retain their bilinear scores; native DAS reports s_{qi}^2. The matched DAS diagnostic changes only this final operation, comparing s_{qi}, s_{qi}^2, and $\text{sign}(s_{qi})s_{qi}^2$ (Appendix C.5).

B.2.3 Additional Benchmark Baselines

Raw pixel similarity. We map RGB pixels to $[0, 1]$ and flatten each image into a feature vector v_a. CIFAR images retain their 32×32 resolution; the 256×256 images are area-downsampled to 64×64 for this baseline. The dot-product score is $v_q^\top v_i$, and the cosine score is $v_q^\top v_i/(\|v_q\|_2\|v_i\|_2)$, with norms clamped away from zero to keep the cosine score well defined during computation.

CLIP similarity. We extract the image embeddings of `openai/clip-vit-base-patch32`, using its image processor for resizing and normalization. Dot-product and cosine attribution scores use the same formulas as the pixel baselines with v_a replaced by the CLIP embedding.

Gradient dot product and cosine. Let ϕ_a^{loss} be the projected feature obtained from gradients of the FM or DDPM squared loss, using the gradient normalization and Monte Carlo averaging above. Both baselines use the final model checkpoint. The dot product retains feature norms, whereas cosine similarity normalizes them to compare feature directions:

$$\tau_{qi}^{\text{dot}} = (\phi_q^{\text{loss}})^\top \phi_i^{\text{loss}}, \qquad \tau_{qi}^{\cos} = \frac{(\phi_q^{\text{loss}})^\top \phi_i^{\text{loss}}}{\|\phi_q^{\text{loss}}\|_2 \|\phi_i^{\text{loss}}\|_2}. \tag{27}$$

TRAK. Our single-model TRAK (Park et al., 2023) adaptation uses ϕ_a^{loss} on both sides of the projected kernel:

$$\tau_{qi}^{\text{TRAK}} = (\phi_q^{\text{loss}})^\top A_\lambda \phi_i^{\text{loss}}, \tag{28}$$

where A_λ is fitted from the training loss-gradient features. D-TRAK replaces the loss readout by the squared output norm, and DAS uses its output readout and terminal square, as specified above.

Relative IF and Renormalized IF. These projected loss-gradient adaptations rescale the TRAK kernel score by a training-sample norm, adjusting its dependence on feature magnitude:

$$\tau_{qi}^{\text{Relative}} = \frac{\tau_{qi}^{\text{TRAK}}}{\|A_\lambda \phi_i^{\text{loss}}\|_2 + 10^{-12}}, \qquad \tau_{qi}^{\text{Renorm}} = \frac{\tau_{qi}^{\text{TRAK}}}{\|\phi_i^{\text{loss}}\|_2 + 10^{-12}}. \tag{29}$$

Relative IF (Barshan et al., 2020) normalizes by the preconditioned training-feature norm; Renormalized IF instead uses the projected feature norm before preconditioning.

TracInCP. At checkpoints c corresponding to 25%, 50%, 75%, and 100% of training, we compute matched training and query loss-gradient features $\phi_{a,c}^{\text{loss}}$. Our TracInCP (Pruthi et al., 2020) implementation takes the unweighted checkpoint mean,

$$\tau_{qi}^{\text{TracInCP}} = \frac{1}{4} \sum_{c=1}^{4} (\phi_{q,c}^{\text{loss}})^\top \phi_{i,c}^{\text{loss}}. \tag{30}$$

GAS. GAS (Hammoudeh & Lowd, 2022) uses the same four checkpoints, replacing each dot product with cosine similarity before averaging. Both feature norms are removed before the four checkpoint contributions are averaged into the final attribution score:

$$\tau_{qi}^{\text{GAS}} = \frac{1}{4} \sum_{c=1}^{4} \frac{(\phi_{q,c}^{\text{loss}})^\top \phi_{i,c}^{\text{loss}}}{\|\phi_{q,c}^{\text{loss}}\|_2 \|\phi_{i,c}^{\text{loss}}\|_2}. \tag{31}$$

Journey-TRAK. Our Journey-TRAK (Georgiev et al., 2023) adaptation computes query features at stored states on the actual generation trajectory. It averages normalized gradients of the mean-output readout over the selected trajectory states, then applies the shared projection. The score is $\tau_{qi} = \phi_q^{\text{journey}\top} A_\lambda \phi_i$, where training features and the kernel use the corresponding output-readout configuration. This target-free mean-readout adaptation is evaluated on generation queries only.

C Sampling Noise, Aggregation, and Score Reweighting

This appendix follows Sections 4.1 and 4.2: score distributions and repeated estimation support the measurement analysis in Figure 1, while aggregation ranges, power transforms, and squaring support the LDS analysis in Figure 2 under matched score and retraining configurations.

C.1 Shared Experimental Settings

Models and data. CIFAR-2 contains 5,000 images, with 2,500 automobile and 2,500 horse images at 32×32 resolution. We use a 36M-parameter unconditional U-Net for linear FM, with model seeds 42, 123, and 456. The DAS DDPM archive uses a separate CIFAR-2 sample and one 38.3M-parameter model. Both settings train for 200 epochs with AdamW, batch size 128, and learning rate 10^{-4}. Generation uses 100-step Euler integration for FM and 50-step DDIM for DDPM. The cross-dataset DAS comparisons use CIFAR-10 FM and ArtBench-2, with settings in Appendix E.1.

LDS evaluation. All empirical LDS values, standard deviations, differences, and confidence intervals are reported on the $\times 100$ scale. The four-method comparisons use 100 generation and 100 class-balanced validation queries per model. Both settings use 64 subsets: FM retains each sample independently with probability 0.5, while DDPM retains exactly half the samples. Query-loss targets average three retraining runs and three noise realizations over 1,000 time points. FM uses $[0.05, 0.95]$ as a proxy for the full-time loss; DDPM uses its 1,000 diffusion steps. FM results report mean and sample standard deviation across model seeds; DDPM intervals use 1,000 paired bootstrap resamples over queries, preserving the method pairing in each draw.

LDS computation and reporting. Keep indicators are converted to deletion indicators by $D = 1 - K$, and the full-model query-loss reference is shared across subsets. LDS uses average ranks for ties and is averaged over the declared queries and model seeds. A constant prediction or response has undefined correlation; we retain that query with recorded value zero and count such cases separately. All aggregation variants use the same score estimates, deletion masks, and measured responses. For DDPM subsets, deleting k samples makes a constant score vector yield identical predictions.

Attribution. FMAS and EK-FAC IF use 250 Monte Carlo draws per training or query gradient, and 125 draws per training image for each curvature-fitting pass. D-TRAK and DAS use 100 time points, one noise draw per image and time point, and projection dimension 4,096. Method constructions are given in Appendix B.2. On CIFAR-2 FM, FMAS uses $\varrho = 10^{-2}$; other scores retain their LDS-selected regularization. Within each setting, transformations reuse gradients, curvature, regularization, queries, and retraining responses, isolating changes to the aggregation rule.

Selection and readout conventions. The main aggregation controls rank samples by absolute pre-square score and retain $k = \lceil \kappa N \rceil$ samples, breaking ties by training index. The same selected indices are used for linear and squared readouts; only their aggregation values change. Native readouts use the square for DAS and linear scores for the other three methods. All comparisons share deletion masks and measured responses. The cross-dataset DAS controls below additionally use signed-support heads, explicitly identified as such; their numerical gains are not interchangeable with those of absolute-score heads. These fixed-fraction controls diagnose aggregation effects; they do not define the noise-based selection rule used by SNR-LDS.

C.2 Score Distributions

For the density plots in Fig. 4 described in Sec. 4.1, we divide each query's scores by $r_q = (N^{-1} \sum_i \tau_{qi}^2)^{1/2}$ before pooling. Each seed contributes $5{,}000 \times 100$ normalized scores. The density uses a shared Gaussian kernel bandwidth across methods and tracks; the plotted curve averages the three seed densities. This rescaling preserves signs and the zero point. It avoids mixing raw scores with different query-specific scales. Magnitude shares are computed separately within each query before averaging. Figure 1(a) averages RMS-normalized score densities over seeds 42, 123, and 456 for all four methods, using DAS before squaring. The marked interval, approximately $[-0.502, 0.502]$, contains half of the pooled normalized FMAS scores.

C.3 Complete-Score Repeats and Sampling Variation

This subsection provides the repeated-score setup and complete results for the sampling-variation analysis in Section 4.1, covering both generation and validation queries.

We compute $R = 16$ independent complete-score estimates for FMAS and D-TRAK on CIFAR-2 seed42, using all 5,000 training samples and the same preselected 16 generation and 16 validation

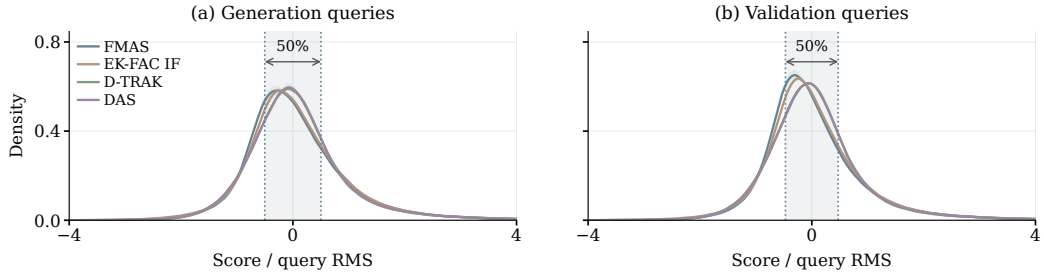


Figure 4: Per-query RMS-normalized score distributions on generation and validation queries. Lines show the mean density across three seeds, and bands show the seed range. The dotted lines in each panel enclose 50% of the corresponding FMAS scores.

queries. Each repeat independently resamples training-side and query-side Monte Carlo draws at the original budgets, while keeping the model, projection, and regularization parameters fixed. FMAS reuses its fitted curvature, while D-TRAK reconstructs its native kernel and normalization from the resampled training features. Thus, the FMAS repeats measure variability conditional on the fitted curvature, not the total variability from refitting every component. For the band-only LDS comparison, an independent pilot defines the magnitude bands. The pilot partitions each query's 5,000 training scores into disjoint absolute-magnitude ranks 0–1%, 1–5%, 5–20%, 20–50%, and 50–100%. FMAS uses 250 Monte Carlo draws on each side; D-TRAK uses its T100 configuration. Table 3 reports relative sampling variation and band-only LDS on these pilot-defined bands. For each query and band, we compute $\mathrm{VarRatio}_{qk}$ from the repeated estimates; the table averages these ratios equally over the 16 queries of each type. Band LDS first averages the 16 independent repeats within each query and then averages the resulting LDS values across queries within each track.

Figure 1(b) uses the same FMAS repeats and 16 generation queries to measure relative sampling variation in finer rank bins. Within each query, it divides all 5,000 samples into ten equal-sized score-rank bins according to $|\bar{\tau}_{qi}|$ and displays them from low to high. For each bin, it reports the ratio between the summed sample variances and the summed squared repeat means, as defined in Equation 5. The plot averages these ratios equally over the 16 generation queries.

In Table 3, the lowest-magnitude half contributes little predictive correlation in all four combinations, while the finer-bin analysis in Figure 1(b) shows substantially greater relative sampling variation towards that region. These repetitions quantify fluctuations around each sample's expected attribution score under fixed estimation settings, not errors against exact LOO effects.

C.4 Power Reweighting

This subsection gives the power-transform settings and results for the score-reweighting analysis in Section 4.2, including both sign-preserving and folded transformations.

For $s_q = \max_i |\tau_{qi}| > 0$, the folded and sign-preserving families are $f_p^{\mathrm{fold}}(\tau_{qi}) = |\tau_{qi}/s_q|^p$ and $f_p^{\mathrm{signed}}(\tau_{qi}) = \mathrm{sign}(\tau_{qi})|\tau_{qi}/s_q|^p$. We hold damping and kernel regularization fixed and evaluate both families at $p = 0, 0.2, \ldots, 4$. Before exponentiation, each query is divided by its maximum absolute score to avoid numerical overflow. This positive query-wise scaling leaves Spearman correlation unchanged. At $p = 0$, the folded family uses unit weights and the signed family retains only signs. Figure 5 reports the four core methods on generation and validation queries, averaging the same three model seeds. DAS starts from its unsquared scores. The sign-preserving family tests continuous changes in relative score weight; the folded family additionally removes sign information. This distinction matters because an ordinary square is not a sign-preserving power transform.

C.5 Score Squaring and Aggregation Across Settings

This subsection supplements the score-squaring and aggregation-range analysis in Section 4.2 with matched-readout comparisons across models, datasets, and query types.

Method	Query	Magnitude ranks	VarRatio	Band LDS ($\times 100$)
FMAS	Gen	0–1%	0.026	47.00
		1–5%	0.048	24.70
		5–20%	0.085	9.97
		20–50%	0.135	0.01
		50–100%	0.444	3.02
FMAS	Val	0–1%	0.024	53.74
		1–5%	0.046	23.88
		5–20%	0.087	13.64
		20–50%	0.162	5.32
		50–100%	0.560	-0.05
D-TRAK	Gen	0–1%	0.003	41.66
		1–5%	0.009	22.91
		5–20%	0.024	10.41
		20–50%	0.070	1.83
		50–100%	0.341	3.48
D-TRAK	Val	0–1%	0.002	47.19
		1–5%	0.008	16.98
		5–20%	0.023	6.48
		20–50%	0.067	6.09
		50–100%	0.320	6.57

Table 3: R16 diagnostics in disjoint bands fixed by an independent pilot, ordered from largest to smallest absolute scores. VarRatio is defined in Equation 5; it and repeat-mean band LDS are averaged equally over queries. Each repeat uses the original scores within a band and zero outside it.

	Original		Squared	
Method	Generation	Validation	Generation	Validation
FMAS	39.37 ± 1.39	44.33 ± 0.27	51.09 ± 0.26	55.80 ± 0.50
EK-FAC IF	38.47 ± 1.45	42.91 ± 0.16	48.24 ± 0.48	53.69 ± 0.96
D-TRAK	39.22 ± 1.49	42.40 ± 0.70	47.35 ± 1.28	53.82 ± 0.09
DAS	38.41 ± 1.05	41.58 ± 0.64	46.94 ± 1.37	52.99 ± 0.17

Table 4: Terminal squaring on CIFAR-2 FM, with all other components fixed. Values are mean$\pm$sample standard deviation across three models. DAS starts before its native square.

We test whether the advantage associated with DAS's native square can be reproduced by changing the readout of other methods, and whether it contracts after low-influence samples are excluded. The relation between output-change and loss-change targets, including the conditional-mean assumption, is derived in Appendix A.4. We first examine the main CIFAR-2 FM setting and then report cross-setting controls that establish the limits of this explanation. For each comparison, the score estimates, deletion subsets, and measured query loss changes remain fixed while the readout or aggregation range varies. The readouts are original scores τ, ordinary squares τ^2, and sign-preserving squares $\mathrm{sign}(\tau)\tau^2$. For DAS, τ denotes the score before its native square. The four-method FM comparison uses original and ordinary-squared scores; the matched DAS and DDPM comparisons also include sign-preserving squares. All LDS values and absolute gains are multiplied by 100.

CIFAR-2 FM: full aggregation. Table 4 reports the results. Squaring increases LDS in all eight method–track comparisons. The gains in the seed means range from 8.13 to 11.72. The original-score spread across methods is 0.96 on generation and 2.75 on validation. Thus, each squaring gain exceeds the between-method spread of the original scores on the corresponding track.

CIFAR-2 FM: matched absolute-score heads. Figure 6 supplements the native-readout comparison in Figure 2(a) with linear readouts for all four methods, using DAS's pre-square scores. For each query, we rank samples by absolute pre-square score and retain 5, 10, 20, 30, 40, 50, 60, 70, 80, 90, or 100%, preserving the selected scores' original signs and values. The curves report generation and validation LDS averaged over three model seeds for each of the four attribution methods.

Panel	Readout	Full LDS	LDS@5%	Fixed-head LDS@5%
C2 Gen	Linear	38.41	45.50	45.50
C2 Gen	Square	46.94	43.86	44.68
C2 Gen	Signed square	45.29	44.68	44.68
C2 Val	Linear	41.58	50.81	50.81
C2 Val	Square	52.99	50.94	51.16
C2 Val	Signed square	51.12	51.16	51.16
C10 Gen	Linear	27.52	35.73	35.73
C10 Gen	Square	39.88	40.49	41.05
C10 Gen	Signed square	40.41	41.05	41.05
C10 Val	Linear	31.59	38.45	38.45
C10 Val	Square	43.37	44.17	44.81
C10 Val	Signed square	44.82	44.81	44.81
AB2 Gen	Linear	24.08	30.12	30.12
AB2 Gen	Square	30.29	31.38	31.49
AB2 Gen	Signed square	31.65	31.49	31.49
AB2 Val	Linear	27.67	34.11	34.11
AB2 Val	Square	37.52	37.80	37.81
AB2 Val	Signed square	37.96	37.81	37.81

Table 5: Matched DAS readouts under each native configuration. CIFAR-2 and CIFAR-10 average three model seeds; ArtBench-2 uses one. Fixed heads are selected from the linear scores. ArtBench-2 linear scores were recovered at the native regularization setting ($\lambda = 1$).

Figure 2(c) compares linear and ordinary-squared readouts on identical sets selected by absolute pre-square score. The score estimates, regularization, queries, deletion masks, and measured responses are fixed. At the top 5%, the squared-minus-linear differences are 0.76, -0.17, 0.00, and 0.05 LDS points for FMAS, EK-FAC IF, D-TRAK, and DAS, respectively. Relative to the full-aggregation gains, these small differences support the interpretation that squaring benefits LDS largely by reducing the relative contribution of low-influence scores. The following experiments examine how far this explanation extends beyond the main setting across further datasets and model settings.

Matched DAS readouts across datasets. The paired comparison uses the scores immediately before squaring under the native DAS configuration, then applies the square once. The model, gradient features, curvature, regularization, queries, and retraining responses are identical in both cases. For each score vector, we compute full-sum LDS and support-direction LDS@5%. We also evaluate both score vectors on the same head selected from the original scores. This control separates changes in membership from reweighting within the selected head, since squaring can promote large negative scores. For each evaluation, the squaring gain is squared-score LDS minus original-score LDS, with paired query-bootstrap intervals reported separately by platform and query track. Tables 5 and 6 report all six platform/query-type panels. Full-sum gains are positive throughout; head gains vary across settings, with a negative ordinary-square gain on CIFAR-2 generation queries and positive gains remaining on both CIFAR-10 tracks and ArtBench-2 validation queries. ArtBench-2 pre-square scores are recovered from existing features at the native $\lambda = 1$.

CIFAR-2 DDPM: matched-query squaring. Table 7 compares the four methods on matched queries. We select damping by original-score LDS on the evaluation queries and hold it fixed for both transforms. The selected values are $\lambda = 10$ for projected methods, $\lambda = 10^{-12}$ for EK-FAC IF, and $\varrho = 10^{-7}/10^{-8}$ for FMAS generation/validation.

Folded squaring improves the projected methods, while the curvature-based methods show no clear gain. For both curvature-based methods, signed squaring exceeds original scores and folded squaring on both tracks; this ordering holds under both damping grids. This exploratory comparison shows that the response to squaring depends on the score construction and its sign information. These matched-query results bound the scope of the FM squaring explanation: neither ordinary squaring nor its benefit after truncation is universal across score constructions and model settings.

Panel	Transformation	Full gain	Head@5% gain	Fixed-head gain
C2 Gen	Square	8.52 [7.32, 9.76]	-1.64 [-2.36, -0.95]	-0.81 [-1.48, -0.17]
C2 Gen	Signed square	6.88 [5.94, 7.84]	-0.81 [-1.48, -0.17]	-0.81 [-1.48, -0.17]
C2 Val	Square	11.41 [9.37, 13.54]	0.13 [-1.28, 1.58]	0.35 [-1.01, 1.70]
C2 Val	Signed square	9.54 [7.74, 11.40]	0.35 [-1.01, 1.70]	0.35 [-1.01, 1.70]
C10 Gen	Square	12.36 [11.02, 13.73]	4.77 [3.87, 5.68]	5.32 [4.51, 6.16]
C10 Gen	Signed square	12.89 [11.83, 14.00]	5.32 [4.51, 6.16]	5.32 [4.51, 6.16]
C10 Val	Square	11.78 [9.82, 13.74]	5.73 [4.18, 7.44]	6.36 [4.80, 8.08]
C10 Val	Signed square	13.23 [11.43, 15.04]	6.36 [4.80, 8.08]	6.36 [4.80, 8.08]
AB2 Gen	Square	6.21 [2.84, 9.45]	1.26 [-0.37, 2.90]	1.37 [-0.06, 2.74]
AB2 Gen	Signed square	7.58 [5.04, 9.98]	1.37 [-0.06, 2.74]	1.37 [-0.06, 2.74]
AB2 Val	Square	9.85 [6.65, 13.09]	3.69 [1.70, 5.68]	3.71 [1.73, 5.69]
AB2 Val	Signed square	10.30 [7.55, 13.22]	3.71 [1.73, 5.69]	3.71 [1.73, 5.69]

Table 6: DAS transformation gains relative to the matching linear readout, with pointwise 95% paired query-bootstrap intervals. Shared validation queries are resampled together across model seeds; generation queries are resampled within each model.

Method	Track	Original	Folded	Signed	Δ [95% CI]
D-TRAK	Gen	24.37	32.50	32.04	$+8.12\ [5.64, 10.44]$
DAS	Gen	23.89	32.12	31.73	$+8.23\ [5.65, 10.59]$
EK-FAC IF	Gen	33.59	31.73	35.48	$-1.86\ [-4.60, 0.57]$
FMAS	Gen	34.30	32.99	35.99	$-1.31\ [-3.98, 1.06]$
D-TRAK	Val	32.72	43.49	44.18	$+10.77\ [8.12, 13.34]$
DAS	Val	32.05	43.05	43.80	$+11.00\ [8.24, 13.81]$
EK-FAC IF	Val	43.31	42.93	45.40	$-0.38\ [-2.79, 1.95]$
FMAS	Val	43.42	43.27	45.54	$-0.15\ [-2.21, 2.08]$

Table 7: Fixed-score transformations on 100 matched DDPM queries per track, including the extended FMAS damping grid. Folded and signed denote τ^2 and $\mathrm{sign}(\tau)\tau^2$; DAS starts from its unsquared score. Δ is folded square minus original LDS.

D SNR-LDS AND SOURCE RETRIEVAL

This appendix gives the implementation and properties of the evaluation methods in Sections 4.3 and 4.4. Dataset-specific settings and evaluation results appear in Appendix E.

D.1 NOISE ESTIMATION AND SNR-BASED AGGREGATION

This subsection specifies the noise fitting, sample selection, aggregation representations, and reporting rules for SNR-LDS introduced in Section 4.3 and used in the experimental setup of Section 5.1.

Zero-mean Gaussian noise fitting. For each method and query, normalize the fitting scores by their RMS r_q, giving $x_{qi} = \widehat{\tau}_{qi}/r_q$. Estimate their density with a Gaussian kernel and bandwidth $h = 0.9\min(\mathrm{SD}, \mathrm{IQR}/1.34)N^{-1/5}$, where SD is the sample standard deviation. Let $a_q = 1.4826\,\mathrm{MAD}(\mathbf{x}_q)$, where MAD is computed about the score median. Use 101 equally spaced points in $[-a_q, a_q]$ to fit the noise-dominated central region. The interval and Gaussian are centered at zero; the median is used only to compute the robust interval width. At the grid points, fit

$$(\widehat{A}_q, \widehat{C}_q) = \arg\min_{A,C} \sum_{x \in \mathcal{G}_q} \widehat{f}_q(x)[\log \widehat{f}_q(x) - A - Cx^2]^2, \qquad \mathcal{G}_q \subset [-a_q, a_q]. \tag{32}$$

There is no linear term because the Gaussian noise mean is fixed at zero. The fitted quadratic coefficient determines the Gaussian width, whereas the intercept controls only its height. For $\widehat{C}_q < 0$, we obtain the normalized scale and convert it back to the original score units:

$$\widehat{\sigma}_{x,q} = \sqrt{-\frac{1}{2\widehat{C}_q}}, \qquad \widehat{\sigma}_q = r_q \widehat{\sigma}_{x,q}. \tag{33}$$

The intercept fits the density height and does not enter selection. No mixture proportion or non-background density is needed. Evaluate $\widehat{\mathrm{SNR}}_{qi} = |x_{qi}|/\widehat{\sigma}_{x,q}$ and retain a sample exactly when this ratio exceeds $\zeta = 3$. Neither noise fitting nor threshold selection uses subset responses.

Fitting and aggregation representations. For methods without a terminal square, the same scores are used for noise fitting and aggregation, as in Equation 7. For DAS in every model setting, fit the noise scale and compute SNR using the signed scores before the native square. Writing these pre-square scores as t_{qi}, the native aggregation is

$$P_{mq}^{\mathrm{SNR,DAS}} = \sum_i d_{mi}\, \mathbf{1}\{|t_{qi}|/\widehat{\sigma}_{t,q} > \zeta\}\, t_{qi}^2. \tag{34}$$

The signed pre-square scores must be obtained from the original construction, since their signs cannot be recovered from squared scores. Both signs are eligible for selection, and the native square is applied once. For paired transformation controls, select from the same original scores and hold the selection fixed while comparing original and transformed aggregation readouts.

Degenerate cases and reporting. Nonpositive bandwidth or central width, rank-deficient or nonfinite fits, $\widehat{C}_q \geq 0$, and nonpositive or nonfinite fitted noise scales are recorded as fitting failures, with SNR-LDS recorded as NA. An all-zero score vector directly yields an empty selected set. A valid fit can also select no scores. Constant predictions or responses follow the existing LDS convention of a recorded value zero, with their counts reported separately. Within-method Full/SNR comparisons use the same valid queries; method-choice utility uses queries valid for every candidate, retaining the complete candidate set. Report fit failures, empty selections, constant predictions, and nonconstant evaluations as separate categories for each method and query track.

D.2 Controlled-Source Retrieval Protocol

This subsection specifies the sources, queries, ranking procedure, and metric aggregation used for the controlled-source retrieval evaluation introduced in Section 4.4.

For each injected concept c, record its source index set $\mathcal{I}_c$ when constructing the training set. The source labels identify images containing the injected concept; the original host-class labels remain unchanged. Queries are generated from the trained model and selected for displaying the concept using the declared detector and human-review procedure, independently of attribution scores. These source and query definitions are shared by all compared methods.

For query q of concept c, rank all N training images in the method's declared support direction, resolving tied scores by training index. Recall@K and AP are defined in Equations 8 and 9, using the injected samples as the relevant set. We first average query results within each concept and then average over concepts. Retrieval searches the full training set. Because these metrics use rankings rather than score sums, strictly increasing score transformations preserve them.

Experimental settings and retrieval results are given in Appendices E.1 and E.5. The protocol complements subset prediction by measuring source recovery without score aggregation.

E Evaluation Experiments

This appendix gives the experimental settings and full results for Section 5. Mechanism experiments are collected in Appendix C, and the SNR-LDS algorithm is specified in Appendix D.1.

E.1 Models, Queries, and Statistical Comparisons

This subsection details the models, queries, subset retraining, and statistical comparisons used in Section 5.1 across the evaluated datasets and generative model settings.

Models and attribution scores. CIFAR-2 contains 2,500 automobile and 2,500 horse images, with an unconditional 36M-parameter U-Net FM model. CIFAR-10 uses all 50,000 images and a class-conditional model of the same architecture, without classifier-free guidance. Both use three

full-model seeds, AdamW for 200 epochs, batch size 128, initial learning rate 10^{-4}, cosine decay, 10% warmup, and random horizontal flips. ArtBench-2 contains 5,000 post-impressionism and ukiyo-e images and uses rank-128 LoRA adaptation of SD3.5 Medium with style-name prompts; its available results use one model seed. The DDPM setting follows Appendix C.5, using the common 100-query comparison. Each benchmark setting uses 100 generation and 100 validation queries. Attribution constructions and sampling budgets are given in Appendix B.2. Actual model coverage is recorded per method: some TracInCP and GAS results use a single model, while pixel and CLIP validation scores can share model-independent inputs.

Subset responses and SNR evaluation. CIFAR FM uses 64 subsets with independent retention probability 0.5 and three retraining runs per subset; ArtBench-2 uses 32 subsets with one run, and DDPM uses 64 fixed-size half-data subsets. CIFAR FM query losses average 1,000 time points in $[0.05, 0.95]$ and three noise realizations; ArtBench-2 uses 100 time points. Full LDS and SNR-LDS use the same scoring configuration, query indices, native aggregation scores, and response array. For DAS in all settings, noise fitting and SNR selection use signed pre-square scores; selected samples contribute their native squared scores. All other methods use their declared native score space for both fitting and aggregation. The noise mean is zero, the noise scale is fitted separately for each method and query, and the default amplitude threshold is $\zeta = 3$.

Keep indicators are converted to deletion indicators by $D = 1 - K$, and each query uses the same full-model loss reference across subsets. Spearman correlation uses average ranks for ties; query-level scores are averaged over the declared queries and model seeds. A constant prediction or response has undefined correlation and is recorded as zero, with these cases counted separately. Noise-scale fits follow the shared rules in Appendix D.1, without using measured responses to choose the aggregation range or SNR threshold. Failed fits are recorded as NA; valid empty selections yield constant predictions and follow the same zero-value reporting convention. Within-method Full/SNR comparisons use the same valid queries; method-selection comparisons use queries valid for every candidate. Existing scoring configurations retain their original LDS-based parameter selection and development-data status. For ArtBench-2 EK-FAC IF, oracle selection maximizes Full LDS on each reported track over ten damping values, yielding 10^{-7} for generation and 10^{-6} for validation. The same track responses are used for selection and evaluation.

Deletion panel. The deletion experiment uses CIFAR-2 seed42 and 50 generation queries, with FMAS, D-TRAK, DAS, and EK-FAC IF as candidates. For each candidate and query, the native top 300 or 1,000 support samples are removed and the model is retrained with the original recipe. Five random deletion sets per budget provide shared controls across queries. Each evaluator selects the candidate with the best benchmark mean and compares its deletion effects on the 50-query panel.

Source queries and retrieval. The CIFAR-10 source experiment replaces 4% of training images while retaining host labels, introducing ten foreign concepts with 200 known sources per concept. The FM query set contains 500 queries, split into 100 validation and 400 test queries, with 10 and 40 per concept. All 13 FM methods are evaluated on the same 400 test queries. The revised DDPM query set contains 100 validation and 400 test queries, with 50 total queries per concept; the test results are reported in Table 2. Concept detection and human review select queries independently of attribution scores; detector-training images are distinct from injected examples. Each method ranks all 50,000 training images to compute global recall@200 and AP. Metrics are averaged within each concept and then equally across concepts. Training index breaks selection ties. FMAS and EK-FAC IF damping are selected using within-host AP on the 100 validation queries. FMAS selects $\varrho = 1$ for both FM and DDPM. EK-FAC IF selects $\lambda = 10^{-9}$ for FM and $\lambda = 10^{-12}$ for DDPM. Parameter selection uses validation queries; values remain fixed for the 400-query test evaluation.

Statistical comparisons. Paired deletion-utility and LDS comparisons use 2,000 query-bootstrap resamples, retaining all methods and budgets for each sampled query. Shared validation-query indices are resampled together across models. Model-seed standard deviations are distinguished from query intervals conditional on a trained model. Retrieval summaries weight concepts equally; their query intervals condition on the corresponding model, source set, and selection procedure.

Method	Delete 300	Delete 1,000
FMAS	0.010238 [0.009126, 0.011427]	0.019353 [0.017455, 0.021416]
D-TRAK	0.008121 [0.007208, 0.009088]	0.013191 [0.012186, 0.014322]
DAS	0.007500 [0.006751, 0.008392]	0.010899 [0.010030, 0.011870]
EK-FAC IF	0.009952 [0.008930, 0.011059]	0.018759 [0.017034, 0.020594]

Table 8: Mean query-loss increases on 50 CIFAR-2 generation queries, with pointwise 95% query-bootstrap intervals. All four methods use their native support rankings.

E.2 SNR Threshold and Reweighting Sensitivity

This subsection gives the threshold and reweighting protocols for the SNR-LDS sensitivity analysis in Section 5.2, specifying the selection rules held fixed across comparisons.

Keep $\zeta = 3$ as the default and report $\zeta \in \{1, 2, 3, 4\}$ without selecting a threshold by the reported LDS. For every threshold, report retained fractions, query coverage, empty selections, Full/SNR paired differences, and method rankings. Use fixed-fraction magnitude heads as reference points; with a shared zero-mean noise scale, a head matched to the selected count has the same membership, apart from boundary ties. For the head and squaring controls, fit once on the original score vector and hold both the noise scale and selection fixed across readouts and head sizes. For the native evaluator comparison, retain the prescribed fitting/aggregation representations in Appendix D.1. Table 12 reports the four core methods at all four thresholds; Figure 3(a) reports the fixed-selection controls.

E.3 Actual Deletion Effects and Evaluation Choices

This subsection reports the deletion-and-retraining results used to assess SNR-LDS in Section 5.2.

Candidate deletion effects. Table 8 reports the measured loss increase for each attribution method on the 50-query panel. The random-deletion means are 0.001837 and 0.002520 for budgets 300 and 1,000; standard deviations across the 250 query/set observations are 0.000961 and 0.000997. All four method means exceed the random-deletion reference at both budgets.

Deletion utility of the best-ranked methods. Original LDS selects DAS with native squared scores, while SNR-LDS selects FMAS. Both evaluators use the same four-candidate valid-query intersections: 83, 75, and 81 generation queries for seeds 42, 123, and 456, respectively, with equal weight on seed means. The resulting Full/SNR benchmark means are 46.70/43.01 for DAS and 39.32/45.04 for FMAS (%). These selection sets are distinct from the fixed 50-query intervention panel. We compare their existing deletion interventions on all 50 generation queries of the seed42 model. For each method and query, its top 300 or 1,000 native-ranked supporting samples are deleted, and the model is retrained under the original recipe. Let $U_{a,q,h}$ denote the measured query-loss increase for method a, query q, and deletion budget h. The mean utility difference is

$$\Delta U_h = \frac{1}{50} \sum_{q=0}^{49} \left[U_{\mathrm{FMAS},q,h} - U_{\mathrm{DAS},q,h} \right]. \tag{35}$$

Table 9 reports the mean utilities, multiplied by 100. FMAS produces 36.49% and 77.56% larger mean loss increases for 300 and 1,000 deletions, respectively. It has a larger deletion effect on 49 of 50 queries at budget 300 and all 50 at budget 1,000. The paired utility differences are 0.002737 [0.002131, 0.003397] and 0.008453 [0.007100, 0.010071] for budgets 300 and 1,000, respectively (pointwise 95% query-bootstrap intervals, unscaled loss units). These results describe one deletion-model seed and the methods selected by the reported benchmark.

Image changes after deletion. For each of the same 50 queries, we regenerate images with the retrained FMAS and DAS deletion models using the original sampling seed (165) and the 100-step Euler sampler. Pixel L2 is the Euclidean distance between flattened images with uint8 pixel values divided by 255. CLIP similarity is the cosine similarity between image embeddings from `openai/clip-vit-base-patch32`. Figure 3(b) reports mean loss changes, L2 distances, and CLIP similarities for both deletion budgets. On this model's 50 queries, pointwise 95% paired

Deleted	LDS (DAS)	SNR-LDS (FMAS)
300	0.75	1.02
1,000	1.09	1.94

Table 9: Evaluator-selected deletion comparison: mean query-loss increase ($\times 100$) for DAS (original LDS) and FMAS (SNR-LDS, $\zeta = 3$). Both methods use the same 50 CIFAR-2 seed42 generation queries and their native deletion rankings at each of the two deletion budgets.

query-bootstrap intervals show larger loss increases and L2 distances, and lower CLIP similarities, for FMAS at both deletion budgets. These metrics describe the change in generated images. Random deletion uses five retrained models per budget, one for each random deletion set. For each metric, we first average the five results for each query and then average over the 50 queries, using the same original images and distance definitions as FMAS and DAS.

Fixed-selection aggregation controls. Figure 3(a) follows the absolute-score head and matched-readout definitions of Figure 2(a,c) on CIFAR-2 generation queries and model seeds 42, 123, and 456. The $\zeta = 3$ zero-mean Gaussian SNR mask is taken from the accepted benchmark evaluation and held fixed across readouts and head sizes. Each absolute-score head is defined from the pre-square scores and intersected with this mask. Panel (a) reports linear and ordinary-squared readouts for all four methods on the identical intersection. Only queries with a valid SNR fit are averaged within each method and seed, after which the three seed means receive equal weight. Expanding the magnitude head from 5% to 100% changes linear SNR-LDS by 0.39, 0.14, 0.01, and 0.20 percentage points for FMAS, EK-FAC IF, D-TRAK, and DAS, respectively; the corresponding squared-readout changes are 0.02, 0.24, 0.17, and 0.14 percentage points. At the full head, squaring changes SNR-LDS by $+0.31$, -0.50, -0.09, and $+0.01$ percentage points in the same method order. Fixed SNR selection reduces sensitivity to both the low-magnitude tail and terminal squaring.

E.4 SNR-LDS Benchmark Results and Coverage

This subsection reports benchmark coverage and additional model results for Section 5.3, distinguishing valid evaluations, empty selections, and noise-fitting failures.

The query-level coverage accounting includes 244 method–seed–track cells and 24,400 query records, excluding the four DDPM TracInCP/GAS entries. Of these, 17,046 have valid noise-scale fits and 7,354 fail the prescribed fit; 3,876 valid records select no samples and receive zero LDS. These physical-record totals include shared validation inputs at each source seed; Table 10 counts model-independent shared validation inputs only once. Each method is averaged over its valid queries, followed by equal averaging across model seeds. Within-method Full/SNR comparisons use identical valid-query sets. Cross-method table entries need not use identical query sets; their differences are descriptive rather than paired significance tests. All pixel/CLIP fits fail under the zero-mean Gaussian rule and are reported as NA, not replaced by historical values or zeros.

Table 11 reports the recomputed ArtBench-2 and CIFAR-10 FM results. DAS is highest on both tracks in both settings; D-TRAK is slightly above FMAS on each CIFAR-10 FM track. DAS fits signed pre-square scores and aggregates selected native squared scores, as specified in Appendix D.1. No cross-method difference in these tables is claimed significant.

E.5 Source Retrieval Results

This subsection provides the injection pairs and per-concept results for the controlled-source retrieval protocol in Section 4.4 and its evaluation in Section 5.4.

Injection pairs and query split. The five fine-grained pairs use related foreign and original concepts; the five coarse-grained pairs use more distinct concepts (Table 13). Each pair replaces 200 training images while retaining their original CIFAR-10 labels. Concept detection and human review select 50 generated queries per pair, split into 10 validation and 40 test queries. Validation queries are used for parameter selection; retrieval ranks all 50,000 training images.

Method	C2 FM		C10 FM		AB2		C2 DDPM	
	Val	Gen	Val	Gen	Val	Gen	Val	Gen
Raw pixel (dot product)	0/100	0/300	0/100	0/300	0/100	0/100	0/100	0/100
Raw pixel (cosine)	0/100	0/300	0/100	0/300	0/100	0/100	0/100	0/100
CLIP similarity (dot product)	0/100	0/300	0/100	0/300	0/100	0/100	0/100	0/100
CLIP similarity (cosine)	0/100	0/300	0/100	0/300	0/100	0/100	0/100	0/100
Gradient (dot product)	300/300	299/300	300/300	300/300	100/100	100/100	90/100	96/100
Gradient (cosine)	300/300	299/300	288/300	279/300	100/100	100/100	25/100	36/100
TracInCP	199/300	204/300	283/300	275/300	100/100	100/100	33/100	45/100
GAS	226/300	235/300	276/300	277/300	100/100	100/100	52/100	68/100
Journey-TRAK	n/a	226/300	n/a	300/300	n/a	100/100	n/a	100/100
Relative IF	300/300	300/300	300/300	300/300	100/100	100/100	100/100	100/100
Renormalized IF	300/300	300/300	300/300	300/300	100/100	100/100	100/100	100/100
TRAK	300/300	300/300	300/300	300/300	100/100	100/100	100/100	100/100
D-TRAK	300/300	300/300	300/300	300/300	100/100	100/100	100/100	100/100
DAS	300/300	300/300	300/300	300/300	100/100	100/100	100/100	100/100
EK-FAC IF	251/300	258/300	286/300	294/300	100/100	100/100	100/100	100/100
FMAS	226/300	249/300	281/300	288/300	100/100	100/100	100/100	100/100

Table 10: Valid noise-scale fits / available queries for the new SNR rule. CIFAR FM counts pool the three seeds; shared pixel/CLIP validation inputs are counted once. Valid empty selections are included in the numerator and assigned zero LDS. All pixel/CLIP queries fail the prescribed noise-scale fit; they are not missing input files.

Method	ArtBench-2		CIFAR-10 FM	
	Validation	Generation	Validation	Generation
Raw pixel (dot product)	NA	NA	NA	NA
Raw pixel (cosine)	NA	NA	NA	NA
CLIP similarity (dot product)	NA	NA	NA	NA
CLIP similarity (cosine)	NA	NA	NA	NA
Gradient (dot product)	21.46	15.52	0.84 ± 0.68	-0.37 ± 0.76
Gradient (cosine)	21.12	15.76	0.00 ± 0.00	-0.04 ± 0.07
TracInCP	22.29	17.38	0.20 ± 0.16	-0.08 ± 0.88
GAS	23.33	17.64	0.00 ± 0.00	0.00 ± 0.00
Journey-TRAK	n/a	29.20	n/a	14.16 ± 1.75
Relative IF	26.81	19.71	27.07 ± 1.04	21.25 ± 1.29
Renormalized IF	27.00	18.13	22.31 ± 0.98	15.75 ± 0.98
TRAK	26.49	17.07	25.35 ± 1.18	18.84 ± 1.40
D-TRAK	36.31	30.01	38.18 ± 0.81	35.67 ± 1.17
DAS	38.48	30.91	44.07 ± 0.43	40.51 ± 0.92
EK-FAC IF	30.39	20.27	35.88 ± 0.80	32.98 ± 0.50
FMAS	27.75	19.67	37.94 ± 0.50	35.56 ± 0.88

Table 11: SNR-LDS (%) with $\zeta = 3$ on ArtBench-2 and CIFAR-10 FM. ArtBench-2 uses one model; CIFAR-10 FM reports means and sample standard deviations across three model seeds. Each model uses 100 queries per track, with means computed over method-specific valid queries. ArtBench-2 EK-FAC IF retains per-track oracle damping selection. NA denotes no valid fits; n/a denotes the unassessed Journey-TRAK validation track, which is outside this method's evaluation scope.

Per-concept results. Tables 14 and 15 report the full 400-query test results for FM and DDPM, respectively. Each concept has 40 test queries; group summaries average the five concept means equally before rounding. For FM, FMAS has mean AP of 0.049 and 0.457 on fine- and coarse-grained pairs, with recall of 0.090 and 0.464. The corresponding AP values for EK-FAC IF are 0.046 and 0.424, with recall of 0.088 and 0.433. D-TRAK leads the fine-grained group, with AP of 0.062 and recall of 0.096.

Setting	Track	Method	$\zeta = 1$	2	3	4
C2 FM	gen	FMAS	41.79	45.39	44.54	41.79
C2 FM	gen	EK-FAC IF	40.36	43.66	43.31	41.28
C2 FM	gen	D-TRAK	40.29	43.58	44.50	42.96
C2 FM	gen	DAS	46.70	45.42	43.73	41.03
C2 FM	val	FMAS	45.89	49.07	49.60	47.24
C2 FM	val	EK-FAC IF	44.33	47.24	48.91	47.66
C2 FM	val	D-TRAK	43.56	46.87	49.05	47.76
C2 FM	val	DAS	52.69	51.87	50.45	48.26
C10 FM	gen	FMAS	32.52	34.27	35.56	35.91
C10 FM	gen	EK-FAC IF	30.71	31.88	32.98	33.82
C10 FM	gen	D-TRAK	28.60	32.07	35.67	38.21
C10 FM	gen	DAS	39.95	40.22	40.51	40.48
C10 FM	val	FMAS	35.25	36.81	37.94	38.66
C10 FM	val	EK-FAC IF	33.23	34.59	35.88	36.83
C10 FM	val	D-TRAK	31.93	35.35	38.18	40.14
C10 FM	val	DAS	43.47	43.86	44.07	44.08
AB2	gen	FMAS	17.24	17.87	19.67	20.10
AB2	gen	EK-FAC IF	16.92	18.12	20.27	19.04
AB2	gen	D-TRAK	27.37	28.10	30.01	30.27
AB2	gen	DAS	30.49	30.76	30.91	31.10
AB2	val	FMAS	25.26	25.79	27.75	29.00
AB2	val	EK-FAC IF	25.92	29.09	30.39	31.55
AB2	val	D-TRAK	30.75	33.99	36.31	34.83
AB2	val	DAS	37.54	38.55	38.48	37.49
C2 DDPM	gen	FMAS	34.74	34.42	34.94	35.92
C2 DDPM	gen	EK-FAC IF	34.04	34.51	34.69	35.18
C2 DDPM	gen	D-TRAK	24.80	27.31	30.38	29.50
C2 DDPM	gen	DAS	31.52	31.42	31.02	29.45
C2 DDPM	val	FMAS	43.62	43.70	44.10	43.96
C2 DDPM	val	EK-FAC IF	43.15	44.13	44.00	45.15
C2 DDPM	val	D-TRAK	32.91	36.02	39.40	40.73
C2 DDPM	val	DAS	41.65	41.56	41.19	39.93

Table 12: SNR-LDS (%) sensitivity for the four core methods. Thresholds share the same fitted noise scale and valid queries within each method. FM means average three model seeds; AB2 and DDPM use one. The default remains $\zeta = 3$, independent of these results.

Group	Injected concept	Original label	Test queries
Fine	leopard	cat	40
Fine	cattle	deer	40
Fine	wolf	dog	40
Fine	camel	horse	40
Fine	tractor	truck	40
Coarse	sunflower	airplane	40
Coarse	castle	automobile	40
Coarse	butterfly	bird	40
Coarse	mushroom	frog	40
Coarse	skyscraper	ship	40

Table 13: CIFAR-100 concepts injected into CIFAR-10 classes. Each pair contributes 200 training images, 10 validation queries, and 40 test queries per model. Original class labels are retained.

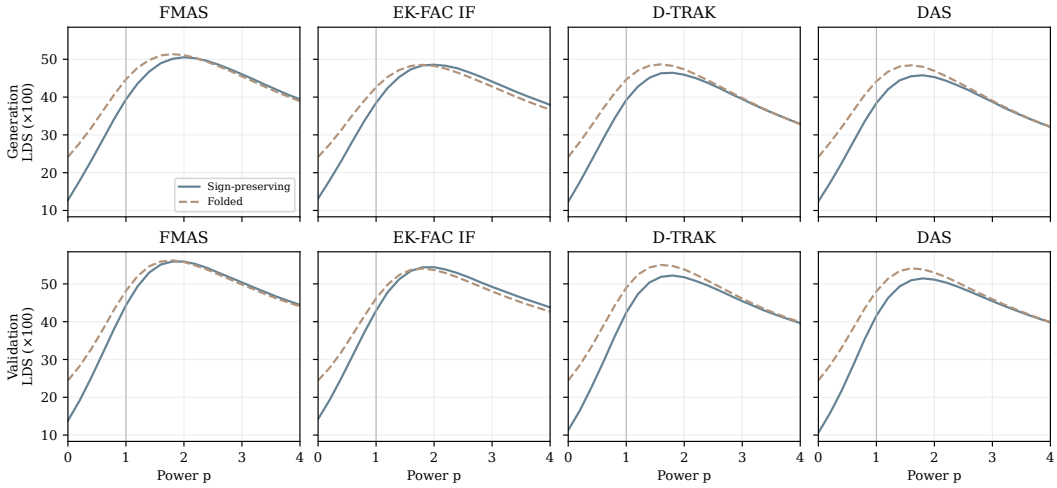


Figure 5: Power reweighting on CIFAR-2 FM generation (top) and validation (bottom) queries. Curves average LDS over three model seeds for sign-preserving and folded transforms with $p = 0, 0.2, \ldots, 4$. DAS uses pre-square scores; scoring settings and subset responses remain fixed.

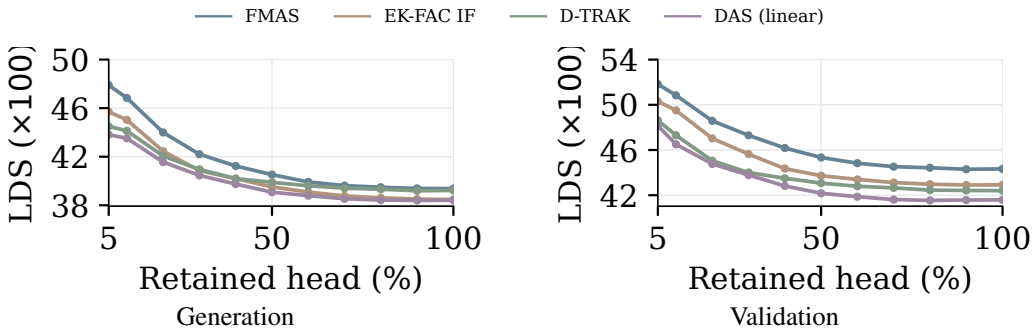


Figure 6: Magnitude-head LDS with linear readouts for all four methods; DAS uses its pre-square scores. Every point is the mean over three model seeds. All grid points are evaluated.

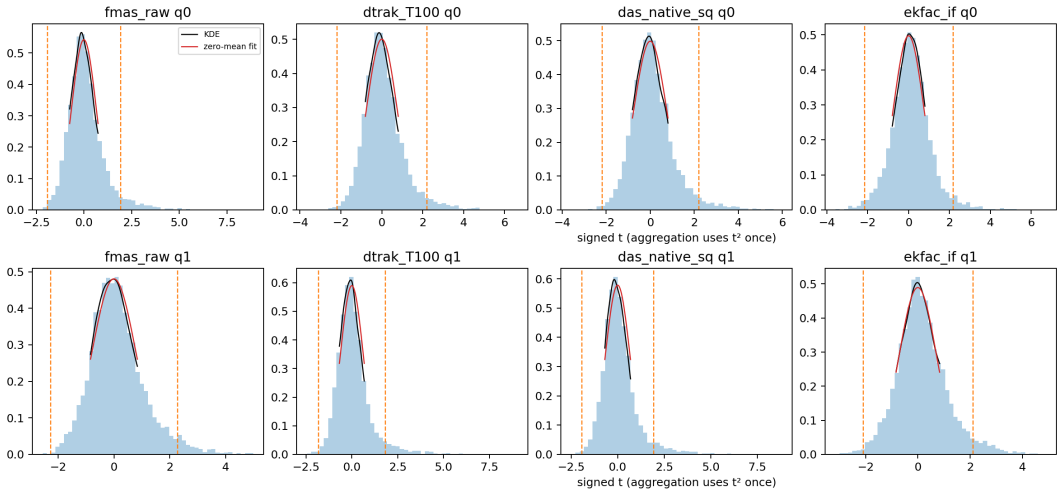


Figure 7: CIFAR-2 FM seed42 queries 0 and 1 under the zero-mean Gaussian SNR evaluator. The four core methods share these predefined generation queries, independently of fit quality and LDS.

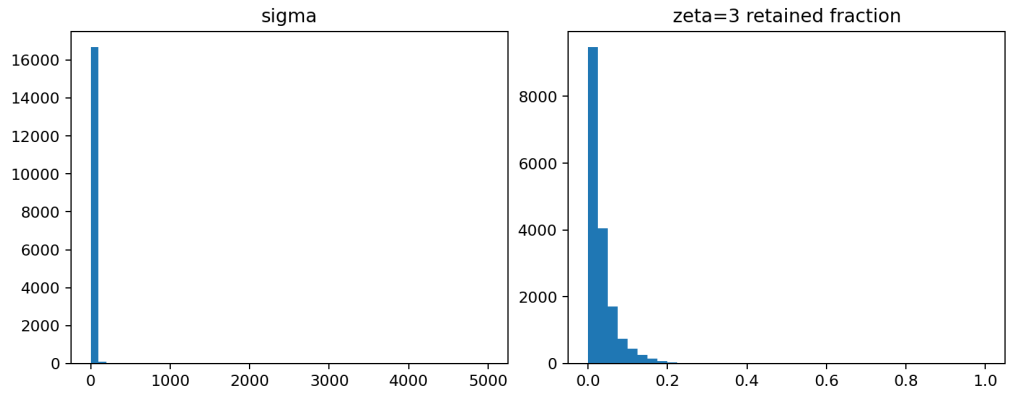


Figure 8: Descriptive distributions of fitted noise scales and retained fractions over valid reevaluated query records. Failed fits are excluded here but included in the reported coverage statistics.

Method	leopard	cattle	wolf	camel	tractor	sunflower	castle	butterfly	mushroom	skyscraper
Global AP										
FMAS	0.031	0.059	0.020	0.063	0.075	0.731	0.926	0.046	0.114	0.467
EK-FAC IF	0.032	0.054	0.021	0.056	0.067	0.721	0.876	0.040	0.090	0.393
D-TRAK	0.093	0.044	0.035	0.055	0.081	0.663	0.861	0.035	0.072	0.409
DAS	0.079	0.036	0.033	0.050	0.066	0.612	0.800	0.034	0.068	0.339
TRAK	0.033	0.017	0.012	0.018	0.028	0.228	0.259	0.012	0.039	0.099
CLIP cosine	0.022	0.056	0.018	0.024	0.038	0.590	0.286	0.025	0.023	0.171
Pixel cosine	0.005	0.004	0.005	0.005	0.006	0.044	0.025	0.004	0.006	0.021
CLIP dot	0.013	0.022	0.013	0.012	0.021	0.380	0.071	0.013	0.009	0.126
Pixel dot	0.003	0.004	0.003	0.005	0.004	0.005	0.007	0.003	0.003	0.006
Gradient dot	0.006	0.005	0.005	0.005	0.007	0.006	0.005	0.005	0.004	0.004
Gradient cosine	0.008	0.005	0.006	0.005	0.008	0.009	0.007	0.006	0.005	0.005
Relative IF	0.040	0.016	0.014	0.020	0.037	0.188	0.273	0.011	0.041	0.086
Renormalized IF	0.025	0.014	0.009	0.014	0.023	0.247	0.216	0.009	0.038	0.086
Recall@200										
FMAS	0.062	0.121	0.050	0.101	0.116	0.699	0.872	0.090	0.175	0.481
EK-FAC IF	0.067	0.111	0.048	0.102	0.114	0.690	0.823	0.079	0.148	0.427
D-TRAK	0.135	0.069	0.061	0.093	0.123	0.630	0.791	0.059	0.107	0.411
DAS	0.118	0.062	0.061	0.093	0.110	0.597	0.736	0.062	0.111	0.366
TRAK	0.072	0.048	0.038	0.055	0.076	0.289	0.307	0.039	0.095	0.163
CLIP cosine	0.032	0.081	0.038	0.035	0.060	0.568	0.325	0.044	0.034	0.211
Pixel cosine	0.007	0.006	0.005	0.007	0.008	0.077	0.055	0.003	0.009	0.043
CLIP dot	0.016	0.041	0.030	0.019	0.040	0.389	0.119	0.021	0.007	0.182
Pixel dot	0.001	0.005	0.000	0.002	0.000	0.002	0.000	0.000	0.000	0.000
Gradient dot	0.003	0.005	0.011	0.006	0.012	0.002	0.010	0.005	0.002	0.003
Gradient cosine	0.015	0.007	0.009	0.008	0.013	0.014	0.014	0.006	0.007	0.008
Relative IF	0.088	0.048	0.044	0.060	0.093	0.252	0.319	0.033	0.099	0.154
Renormalized IF	0.058	0.043	0.025	0.046	0.065	0.299	0.276	0.028	0.093	0.146

Table 14: Per-concept FM retrieval on 400 test queries. The first five concepts are fine-grained; the remaining five are coarse-grained. All methods rank the full 50,000-image training set.

	Fine-grained					Coarse-grained				
Method	leopard	cattle	wolf	camel	tractor	sunflower	castle	butterfly	mushroom	skyscraper
					Global AP					
FMAS	0.070	0.043	0.023	0.081	0.119	0.430	0.821	0.083	0.165	0.469
EK-FAC IF	0.070	0.037	0.020	0.072	0.108	0.349	0.770	0.065	0.147	0.440
D-TRAK	0.190	0.050	0.039	0.077	0.121	0.755	0.948	0.062	0.082	0.590
DAS	0.170	0.047	0.037	0.066	0.093	0.722	0.856	0.050	0.068	0.510
TRAK	0.052	0.019	0.012	0.020	0.033	0.209	0.254	0.018	0.042	0.131
CLIP cosine	0.029	0.062	0.020	0.032	0.040	0.472	0.225	0.025	0.018	0.176
Pixel cosine	0.005	0.005	0.004	0.005	0.005	0.038	0.024	0.004	0.006	0.014
CLIP dot	0.015	0.024	0.014	0.013	0.020	0.266	0.046	0.013	0.008	0.134
Pixel dot	0.003	0.004	0.003	0.005	0.004	0.005	0.007	0.003	0.003	0.006
Gradient dot	0.005	0.004	0.005	0.005	0.006	0.014	0.006	0.006	0.004	0.005
Gradient cosine	0.005	0.004	0.004	0.005	0.006	0.020	0.006	0.006	0.004	0.005
Relative IF	0.064	0.018	0.014	0.022	0.041	0.160	0.276	0.015	0.038	0.107
Renormalized IF	0.032	0.013	0.008	0.010	0.020	0.149	0.152	0.010	0.029	0.083
					Recall@200					
FMAS	0.127	0.095	0.061	0.133	0.169	0.456	0.788	0.136	0.223	0.493
EK-FAC IF	0.127	0.086	0.054	0.135	0.163	0.387	0.745	0.115	0.208	0.477
D-TRAK	0.240	0.068	0.058	0.111	0.165	0.709	0.889	0.093	0.110	0.570
DAS	0.202	0.062	0.055	0.099	0.129	0.685	0.795	0.073	0.094	0.494
TRAK	0.099	0.056	0.035	0.059	0.079	0.275	0.305	0.051	0.100	0.199
CLIP cosine	0.049	0.088	0.034	0.053	0.066	0.465	0.260	0.044	0.025	0.215
Pixel cosine	0.004	0.007	0.003	0.008	0.004	0.071	0.051	0.004	0.009	0.029
CLIP dot	0.022	0.043	0.030	0.023	0.036	0.292	0.088	0.022	0.005	0.181
Pixel dot	0.000	0.005	0.000	0.003	0.000	0.002	0.000	0.000	0.000	0.000
Gradient dot	0.001	0.004	0.006	0.009	0.006	0.005	0.006	0.008	0.003	0.005
Gradient cosine	0.005	0.004	0.005	0.006	0.009	0.026	0.011	0.008	0.006	0.005
Relative IF	0.123	0.053	0.039	0.064	0.096	0.230	0.320	0.043	0.095	0.179
Renormalized IF	0.073	0.037	0.021	0.028	0.050	0.220	0.230	0.030	0.072	0.146

Table 15: Per-concept DDPM retrieval on the 400-query test set, with 40 queries per concept. All methods rank the full 50,000-image training set; tuning uses 100 separate validation queries.

F NOTATION INDEX

The tables summarize notation used in the paper and provide links to its definitions. Bold symbols denote vectors or matrices, hats denote computed estimates, and a superscript $*$ denotes trained parameters. Context distinguishes overloaded symbols.

Symbol	Meaning	Definition
Data, models, and losses		
$\mathbb{S}$, N	Training set and its size.	Sec. 3.1
$\mathbf{z}^{(i)} = (\mathbf{x}^{(i)}, y^{(i)})$	Training image and its class label or text condition.	Sec. 3.1
i, q	Training-sample and query indices.	Sec. 3.2
$\mathbf{x}^q$, $\mathcal{Q}$, N_q	Query image, query set, and number of queries.	Sec. 3.3
$\mathbf{x}_0$, t, $\mathbf{x}_t$	Gaussian noise, interpolation time, and interpolated image in Flow Matching.	Sec. 3.1
v_θ, o_θ	FM velocity field; generic FM velocity or DDPM noise output.	Sec. 3.1; App. B.2
θ, P_θ	Trainable parameter vector and its dimension.	App. A.1
θ^*, $\theta^*_{\setminus i}$	Full-data parameters and parameters after deleting sample i and retraining.	Sec. 3.2
ℓ_i, $\mathcal{L}_\mathbb{S}$	Per-sample loss and mean training loss.	Sec. 3.2
$\mathcal{J}_q$, $\widehat{\mathcal{J}}_q$	Expected query loss and its finite-sampling measurement.	Sec. 3.3
D (output dimension)	Number of output coordinates; implementation losses average over these coordinates.	App. A.2
Attribution and curvature		
Δ_{qi}, $\boldsymbol{\Delta}_q$	LOO query-loss change and its vector over all training samples.	Sec. 3.2
$\widehat{\tau}_{qi}$, $\widehat{\boldsymbol{\tau}}_q$	Computed attribution score and query-specific score vector in $\mathbb{R}^N$.	Sec. 3.2
τ^{IF}_{qi}	First-order influence-function score.	Eq. 2
H, $\widehat{H}$	Training-loss Hessian and estimated curvature used for attribution.	Sec. 3.2
g_a, $\widehat{g}_a$, m_a	Gradient, Monte Carlo gradient estimate, and sampling budget for image a.	App. A.2
J_a, u_a, r_a	Output Jacobian, target velocity, and output residual.	App. A.2
H_{GGN}	Generalized Gauss–Newton curvature.	App. A.2
Q_b, Λ_b	EK-FAC eigenbasis and eigenvalues for parameter block b.	App. A.2
$\widetilde{g}_{a,b}$	Gradient estimate expressed in block b's eigenbasis.	App. A.2
δ, δ_l, ϱ	FMAS damping matrix, layer damping, and dimensionless damping multiplier.	Eq. 3
$\lambda_{\mathrm{global}}$	Global damping used by EK-FAC IF.	App. B.2
$h_a^{(k)}$, $\bar{h}_a$, T	Readout gradient for draw k, averaged normalized gradient, and number of draws.	App. B.2
R_{proj}, p, P_{all}	Random projection, projected dimension, and number of parameters before projection.	App. B.2
ϕ_a, G	Projected feature and matrix of training features.	App. B.2
K_λ, A_λ	Regularized feature Gram matrix and its rescaled inverse.	App. B.2
s_{qi}	Bilinear score before a terminal transformation; native DAS squares this value.	App. B.2
LDS and estimation errors		
M, $\mathbb{S}_m$, θ^*_m	Number of retraining subsets, retained subset m, and its retrained parameters.	Sec. 3.3
α	Independent training-sample retention probability.	Sec. 3.3
d_{mi}, $\mathbf{d}_m$	Deletion indicator and deletion vector for subset m.	Sec. 3.3
K, $D = 1 - K$ (subset matrices)	Keep and deletion indicator matrices; distinct from K_λ and output dimension D.	App. E.1
P_{mq}, Y_{mq}	Predicted and measured query-loss changes for subset m.	Sec. 3.3
ρ_S, LDS_q	Spearman correlation and LDS for query q.	Eq. 4
ω, $\boldsymbol{\eta}_q$	Score-estimation randomness and attribution signal $\mathbb{E}_\omega[\widehat{\boldsymbol{\tau}}_q]$.	Sec. 4.1

Symbol	Meaning	Definition
$\boldsymbol{\varepsilon}_q$	Zero-mean sampling error $\widehat{\boldsymbol{\tau}}_q - \boldsymbol{\eta}_q$.	Sec. 4.1
Σ_q	Covariance of the query-specific score-estimation noise.	App. A.3
$\mathbf{w}_{mn} = \mathbf{d}_m - \mathbf{d}_n$	Difference of deletion vectors for subsets m and n.	Eq. 19
Aggregation and repeated-score diagnostics		
s_q, p (power)	Maximum absolute score and power-transformation exponent.	Sec. 4.2
κ, $k = \lceil \kappa N \rceil$	Retained fraction and head size in aggregation diagnostics.	App. C.1
R, $\bar{\tau}_{qi}$, s_{qi} (repeats)	Number of independent estimates, repeat mean, and repeat sample standard deviation.	App. C.3
B_{qk}, $\mathrm{VarRatio}_{qk}$	Query-specific magnitude bin and its relative sampling variation.	Eq. 5
Signal-to-Noise Ratio LDS		
r_q, $\mathbf{x}_q$ (normalization)	Query-wise score RMS and normalized fitting scores.	Sec. 4.3
σ_q, $\widehat{\sigma}_q$	Noise standard deviation and its estimate in original score units; noise mean is zero.	Sec. 4.3; Eq. 33
$\widehat{\sigma}_{x,q}$	Fitted noise standard deviation in RMS-normalized score units.	Eq. 33
SNR_{qi}, $\widehat{\mathrm{SNR}}_{qi}$	Population and empirical amplitude signal-to-noise ratios.	Sec. 4.3
ζ	SNR selection threshold; default 3, sensitivity values $1, 2, 3, 4$.	Sec. 4.3
H_q, $\widetilde{\tau}_{qi}$	SNR-selected training indices and scores set to zero outside the selection.	Sec. 4.3
P^{SNR}_{mq}	Subset prediction from SNR-selected scores.	Eq. 7
$\widehat{f}_q$, h, a_q, $\mathcal{G}_q$	KDE, bandwidth, central half-width, and zero-centered fitting grid.	App. D.1
$\widehat{A}_q$, $\widehat{C}_q$	Intercept and quadratic coefficient of the zero-mean Gaussian log-density fit.	Eq. 32
t_{qi} (DAS fitting)	Signed pre-square DAS score; selected samples contribute t_{qi}^2.	Eq. 34
Source retrieval and deletion evaluation		
c, α_{inj}, $\mathcal{I}_c$	Injected concept, replacement fraction, and injected training indices.	Sec. 4.4
K (retrieval), $\mathcal{R}_q(K)$	Retrieval cutoff and top-K training indices for query q.	Eq. 8
z_{qj}	Indicator that the sample at rank j is an injected source.	Eq. 9
$\mathrm{Recall}_q@K$, AP_q	Fraction of injected sources recovered in the top K and full-ranking average precision.	Sec. 4.4
$U_{a,q,h}$, ΔU_h	Loss increase after deleting h samples ranked by method a, and mean FMAS-minus-DAS utility difference.	Eq. 35