A Compass for Useful Data: Online Data Selection via Alignment-Gated Fisher Geometry

Jingwen Zhang[1,2*] **Jianrui Shi**[2,3*] **Zheng Wang**[2]
Wanfang Chen[2] **Yaping Wang**[1†] **Kaixuan Zhang**[2†]
[1]KLATASDS-MOE, School of Statistics,
East China Normal University
[2]Corporate Research, Bosch (China) Investment Co., Ltd.
[3]National University of Singapore

Abstract

Online data selection is typically driven by per-sample scores such as loss, uncertainty, or gradient norm, which measure how strongly the model reacts to a candidate but not what parameter update the candidate would induce. What an online learner should select is therefore not the most striking example, but the one whose update is most worth taking. To make this concrete, we score a candidate by the movement induced by its update, using a Fisher-whitened information gain that evaluates the update in the loss geometry rather than by raw gradient size. Movement, however, is not yet progress, so we pass this score through a *Descent Alignment Gate*, a soft factor that retains it only when the induced update agrees with the descent direction. This yields a sample-level score favoring updates that are both informative and descent-aligned. To make the criterion compatible with training, we extend it from samples to batches, where selected updates should not only score well individually but also complement one another. We capture this with the *Gated Fisher Volume*, a log-determinant batch objective that reduces to the per-sample score on a singleton, grows with the volume spanned by the chosen updates in the Fisher geometry, and admits a greedy algorithm with a constant-factor approximation guarantee. Across supervised fine-tuning benchmarks, our selector consistently outperforms online selection baselines under matched training budgets. More broadly, the framework reframes data selection itself: rather than a spotlight on conspicuous examples, it acts as a compass toward updates worth taking.

1 Introduction

Fine-tuning large language models under limited training budgets often requires selecting a small but useful subset of training examples [1, 50]. During training, data selection and model state form a feedback loop: selected samples determine the next update, and the updated model state changes how future candidates should be evaluated. This coupling motivates online data selection, where candidate scores adapt to the current training state throughout fine-tuning [31, 45, 52].

Many online selection methods measure sample value through the current model's response to each sample. Loss-based methods prioritize examples that the model currently fits poorly [31, 22], while importance-sampling methods use gradient magnitude as a proxy for training effect [23]. Other methods use reducible loss, training dynamics, or smoothed loss signals to identify useful examples during training [38, 29]. Recent LLM-focused online selectors further incorporate Taylor-style

*Equal contribution.
†Corresponding authors: Yaping Wang (ypwang@fem.ecnu.edu.cn) and Kaixuan Zhang (kaixuan.zhang@cn.bosch.com).

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

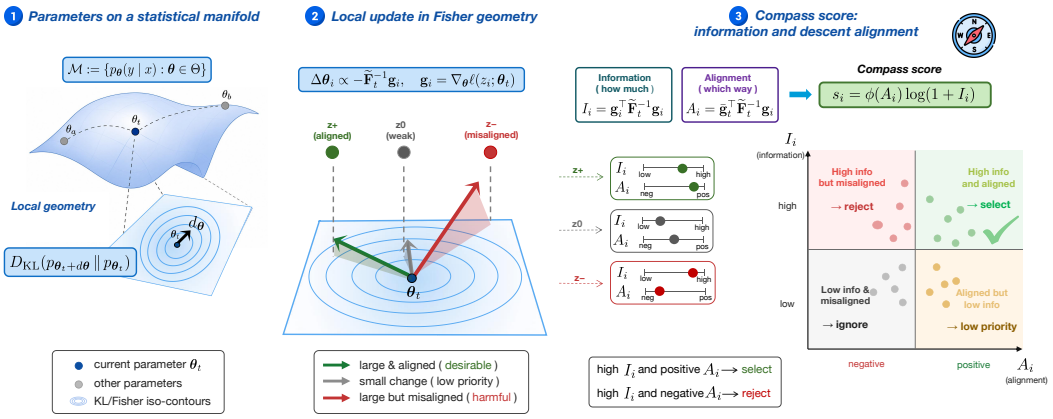


Figure 1: **Overview of Alignment-Gated Fisher Geometry.** Candidates are evaluated by the parameter updates they induce in the model's local Fisher geometry. The Fisher-whitened information I_i measures update magnitude, while the descent alignment A_i measures agreement with a reference descent direction. The compass score $s_i = \phi(A_i)\log(1 + I_i)$ prioritizes informative, descent-aligned candidates and suppresses low-information or misaligned ones.

utilities and diversity-aware batch criteria [45, 52]. These signals identify salient candidates under the current model, but a selected sample affects training through the parameter update. Its value therefore depends on how that update changes the current model state. We shift online data selection from response scoring to parameter-update evaluation.

We instantiate this view through *Alignment-Gated Fisher Geometry*, shown in Figure 1. At each training step, candidate is evaluated by the movement its update induces in parameter space. We measure this movement in Fisher geometry, where local parameter displacement corresponds to distributional change in the predictive model [3, 35]. This gives the Fisher-whitened information score I_i. Since movement alone can conflict with descent, we gate this information by the Fisher-aligned reference direction A_i, yielding $s_i = \phi(A_i)\log(1 + I_i)$.

We further extend the sample-level rule to batch selection through *Gated Fisher Volume* (GFV). At the batch level, individually high-scoring samples may induce redundant Fisher-whitened updates. GFV addresses this with a calibrated log-determinant objective: the calibration preserves the sample score on singleton sets, while the determinant rewards complementary directions and discounts overlap. This yields an efficient greedy selection rule with a constant-factor approximation guarantee. Across fine-tuning benchmarks under matched budgets, our GFV selector achieves better performance across the evaluated backbones. Ablations show that Fisher-whitened information, descent alignment, and batch-level nonredundancy provide complementary gains. Selection geometry diagnostics further confirm that GFV turns a high-score candidate pool into a less redundant update batch.

This work shows that useful data are defined by the updates they induce in parameter space. GFV turns this principle into a practical online batch selector, enabling limited fine-tuning budgets to focus on updates that are informative, descent aligned, and nonredundant within a batch.

2 Background

2.1 Online Data Selection

Let $\mathcal{D}$ denote a large training corpus and let $p_{\boldsymbol{\theta}}(y \mid x)$ be the predictive model at parameters $\boldsymbol{\theta}$. Online data selection asks which examples to use for the next optimization step, rather than which are globally valuable a priori [31]. At iteration t, the learner is at state $\boldsymbol{\theta}_t$ and observes a candidate pool $\mathcal{P}_t = \{z_i = (x_i, y_i)\}_{i=1}^{M} \subset \mathcal{D}$. Given a batch budget $B \ll M$, the goal is to select a subset $S_t \subseteq \mathcal{P}_t$ with $|S_t| = B$ that is most useful under the current model state:

$$S_t \in \arg\max_{S \subseteq \mathcal{P}_t,\ |S|=B} U_t(S; \boldsymbol{\theta}_t).$$

Here U_t is a state-dependent batch utility. Recent LLM-focused online selectors instantiate this idea by updating utility estimates throughout fine-tuning [45]. A common simplification is to score each candidate independently and use an additive objective,

$$U_t^{\mathrm{add}}(S;\boldsymbol{\theta}_t) = \sum_{z_i \in S} s_t(z_i;\boldsymbol{\theta}_t),$$

which reduces selection to choosing the top-B samples by a per-sample score. This is computationally attractive but ignores both the state-dependence of sample value and the redundancy among individually high-scoring examples.

2.2 Statistical Manifold and Fisher Geometry

Evaluating parameter updates requires a geometry on model space. We view the predictive family $\mathcal{M} := \{p_{\boldsymbol{\theta}}(y \mid x) : \boldsymbol{\theta} \in \Theta\}$ as a statistical manifold and equip it with the Fisher information matrix [13, 39, 4]

$$\boldsymbol{F}_t := \mathbb{E}_{(x,y)\sim p_{\boldsymbol{\theta}_t}}\left[\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}_t}(y \mid x)\, \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}_t}(y \mid x)^\top\right],$$

which induces the local KL metric [3, 35]

$$D_{\mathrm{KL}}\big(p_{\boldsymbol{\theta}_t + d\boldsymbol{\theta}} \,\|\, p_{\boldsymbol{\theta}_t}\big) = \frac{1}{2} d\boldsymbol{\theta}^\top \boldsymbol{F}_t\, d\boldsymbol{\theta} + o(\|d\boldsymbol{\theta}\|^2).$$

Thus $\boldsymbol{F}_t$ measures the distributional cost of a parameter perturbation; directions that appear large in Euclidean coordinates may be small in predictive effect, and vice versa. In practice we use the damped variant $\widetilde{\boldsymbol{F}}_t := \boldsymbol{F}_t + \epsilon \boldsymbol{I}$ ($\epsilon > 0$) to ensure invertibility. This Fisher metric underpins the constructions in Sections 3 and 4.

3 Information Gain for Online Sample Selection

We now derive the sample-level score used by our online selection rule. The construction proceeds in three steps. We first identify parameter-space information gain as the first-principles notion of sample utility. We then derive a lightweight local approximation that makes this quantity computable. Finally, we retain only the part of this information that is consistent with optimization progress, yielding the alignment-gated selection rule used throughout the paper.

3.1 Why Parameter-Space Information Gain?

At iteration t, a candidate sample $z_i = (x_i, y_i)$ matters only through the parameter update it would induce. Common selection heuristics (loss, predictive entropy, gradient norm) measure how strongly the model reacts to a sample, but not whether it produces a meaningful change in model state.

For online data selection, the relevant object is the current belief over parameters. We therefore ask a Bayesian design question: if the current local belief is $p_t(\boldsymbol{\theta})$, how much would this belief change after observing z_i? Let $p_{t+1}^{(i)}(\boldsymbol{\theta})$ denote the posterior obtained by incorporating only sample i into the current local posterior. We define the realized parameter-space information gain as

$$\mathrm{IG}_i := D_{\mathrm{KL}}\Big(p_{t+1}^{(i)}(\boldsymbol{\theta}) \,\|\, p_t(\boldsymbol{\theta})\Big). \tag{1}$$

This quantity measures how much sample i would change the local parameter belief. Unlike loss or uncertainty, it evaluates the sample through its induced posterior displacement rather than the model's immediate prediction error. This makes parameter-space information gain a direct proxy for whether the sample would meaningfully change the current model state.

3.2 Local Gaussian Information Gain

We now turn the first-principles quantity in (1) into a computable score. Since the exact posterior change is intractable online, we work in a local approximation around the current iterate $\boldsymbol{\theta}_t$ and model the current posterior as [21]

$$p_t(\boldsymbol{\theta}) \approx \mathcal{N}(\boldsymbol{\theta}_t, \boldsymbol{\Sigma}_t). \tag{2}$$

For sample i, let

$$\ell_i(\boldsymbol{\theta}) := \ell(\boldsymbol{\theta}; z_i), \qquad \boldsymbol{g}_i := \nabla_{\boldsymbol{\theta}} \ell_i(\boldsymbol{\theta}_t), \qquad \boldsymbol{H}_i := \nabla^2_{\boldsymbol{\theta}} \ell_i(\boldsymbol{\theta}_t). \tag{3}$$

Using the standard second-order local approximation around $\boldsymbol{\theta}_t$,

$$\ell_i(\boldsymbol{\theta}) \approx \ell_i(\boldsymbol{\theta}_t) + \boldsymbol{g}_i^\top (\boldsymbol{\theta} - \boldsymbol{\theta}_t) + \frac{1}{2} (\boldsymbol{\theta} - \boldsymbol{\theta}_t)^\top \boldsymbol{H}_i (\boldsymbol{\theta} - \boldsymbol{\theta}_t),$$

the sample-updated posterior remains Gaussian under the same local model:

$$p^{(i)}_{t+1}(\boldsymbol{\theta}) \propto p_t(\boldsymbol{\theta}) \exp\{-\ell_i(\boldsymbol{\theta})\} \tag{4}$$

is again Gaussian, with

$$\boldsymbol{\Sigma}^{(i)}_{t+1} = (\boldsymbol{\Sigma}_t^{-1} + \boldsymbol{H}_i)^{-1}, \qquad \boldsymbol{\mu}^{(i)}_{t+1} = \boldsymbol{\theta}_t - (\boldsymbol{\Sigma}_t^{-1} + \boldsymbol{H}_i)^{-1} \boldsymbol{g}_i. \tag{5}$$

The full derivation is deferred to Appendix B; here we state only the consequence needed for ranking.

Theorem 1 (Local proxy for parameter-space information gain)**.** *Under the local Gaussian approximation* (2)–(5) *and the Gauss–Newton reduction $\boldsymbol{H}_i \approx \boldsymbol{g}_i \boldsymbol{g}_i^\top$, define $r_i := \boldsymbol{g}_i^\top \boldsymbol{\Sigma}_t \boldsymbol{g}_i$. Then the realized information gain IG_i is governed by the scalar r_i. In particular, it is monotone in r_i and admits the lightweight proxy*

$$\mathrm{IG}_i \approx \frac{1}{2} \log(1 + r_i). \tag{6}$$

Consequently, ranking samples by local information gain is equivalent, at this order, to ranking them by r_i or by $\log(1 + r_i)$.

The quantity r_i is the gradient energy in the posterior covariance geometry: it measures how strongly sample i perturbs the current posterior in local parameter coordinates, bridging the first-principles definition (1) to a practical ranking criterion.

To make r_i computable online, we approximate the posterior covariance by the inverse of the damped Fisher matrix $\widetilde{\boldsymbol{F}}_t$ introduced in Section 2.2. Under $\boldsymbol{\Sigma}_t \approx \widetilde{\boldsymbol{F}}_t^{-1}$, Theorem 1 yields the practical information score

$$I_i := \boldsymbol{g}_i^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i, \qquad \mathrm{IG}_i \approx \frac{1}{2} \log(1 + I_i). \tag{7}$$

The sample utility thus reduces to a quadratic form in the Fisher metric followed by logarithmic compression $\psi(I_i) := \log(1 + I_i)$.

3.3 Descent Alignment Gating and the Final Sample Score

The information score captures the magnitude of the parameter-space change a sample induces, but magnitude alone is insufficient: a large update that points against the descent direction harms training regardless of its informativeness. A useful selection rule must therefore ensure that the induced update is consistent with optimization progress.

We formalize descent consistency through the natural-gradient update[3] [3] $\Delta\boldsymbol{\theta}_i = -\eta \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i$. Let $\mathcal{L}(\boldsymbol{\theta})$ be the training objective with $\boldsymbol{G}_t := \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_t)$. A first-order expansion gives

$$\mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_i) - \mathcal{L}(\boldsymbol{\theta}_t) = -\eta\, \boldsymbol{G}_t^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i + O(\eta^2),$$

so the first-order loss change is governed by the sign of

$$A_i^* := \boldsymbol{G}_t^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i. \tag{8}$$

If $A_i^* > 0$ the sample decreases $\mathcal{L}$ to first order; if $A_i^* < 0$ it increases $\mathcal{L}$, regardless of information magnitude.

[3]We use $\widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i$ rather than $\boldsymbol{g}_i$ so that alignment is measured in the same Fisher geometry that defines I_i; otherwise a sample could appear Euclidean-aligned yet be Fisher-misaligned.

Since $\boldsymbol{G}_t$ is unavailable online, we replace it by a reference gradient $\bar{\boldsymbol{g}}_t$, such as the candidate-pool average or an exponential moving average of recent gradients.

We therefore gate the information term by descent alignment and define the sample-level rule as

$$s_i := \phi(A_i)\log(1+I_i), \qquad I_i := \boldsymbol{g}_i^\top \widetilde{\boldsymbol{F}}_t^{-1}\boldsymbol{g}_i, \qquad A_i := \bar{\boldsymbol{g}}_t^\top \widetilde{\boldsymbol{F}}_t^{-1}\boldsymbol{g}_i. \tag{9}$$

Equation (9) is the sample-level rule used throughout the paper. The term $\log(1+I_i)$ measures how informative the sample is in parameter space, while $\phi(A_i)$ retains the information that is consistent with descent.

The following result formalizes this necessity by showing that a rule based only on parameter-space information can select updates that are locally inferior to positively aligned alternatives under the first-order loss expansion.

Theorem 2 (Necessity of descent alignment gating)**.** *Assume the candidate pool contains enough positively aligned samples to form a batch of size B. Any information-only selected batch that contains some $A_i^* < 0$ is, for sufficiently small η, strictly worse in first-order loss than a feasible positively aligned batch. A formal statement and proof are given in Appendix C.2.*

The gate is therefore a necessary mechanism, since information alone can favor samples whose induced updates oppose the objective.

4 From Sample Scores to Batch Selection

Section 3 assigns each candidate a sample-level score, but online training selects a *batch*. At the batch level, the central challenge shifts from utility to *redundancy*: individually strong samples may correspond to nearly the same update direction and provide little additional value when selected together. This section lifts the sample-level criterion to a batch objective that preserves singleton meaning while favoring complementary Fisher-whitened update directions.

4.1 Calibrated Gated Fisher Volume

The sample score already quantifies individual utility; what remains is to model *interaction*, namely whether two individually good samples are complementary or redundant. Let $\boldsymbol{u}_i := \widetilde{\boldsymbol{F}}_t^{-1/2}\boldsymbol{g}_i$, so that $I_i = \|\boldsymbol{u}_i\|_2^2$. Here $\boldsymbol{u}_i$ is the Fisher-whitened update direction associated with sample i, and its squared norm recovers the sample-level information quantity I_i already defined in Section 3. To extend the singleton score $s_i = \phi(A_i)\log(1+I_i)$ to a set S, we want two properties: if $S=\{i\}$, the batch score should reduce to s_i; if selected samples point in distinct whitened directions, their contributions should accumulate rather than cancel. A log-determinant provides a natural batch-level measure of Fisher-whitened update volume, $\log\det(\boldsymbol{I} + \sum_{i\in S}\beta_i \boldsymbol{u}_i\boldsymbol{u}_i^\top)$, discounting overlap when samples share update directions and accumulating contributions when they span orthogonal directions.

To preserve singleton consistency, we calibrate each coefficient so that $|S|=1$ recovers the original sample score s_i:

$$\alpha_i := \frac{\exp(s_i)-1}{I_i} = \frac{(1+I_i)^{\phi(A_i)}-1}{I_i}, \tag{10}$$

and for $I_i = 0$ we set $\alpha_i = \phi(A_i)$ by continuity. The calibrated weights α_i are nonnegative by construction.

The resulting batch-level objective is

$$J_{\mathrm{GFV}}(S) := \log\det\left(\boldsymbol{I} + \sum_{i\in S}\alpha_i \boldsymbol{u}_i\boldsymbol{u}_i^\top\right). \tag{11}$$

We call Eq. (11) the *Gated Fisher Volume* (GFV) score.

4.2 Properties and Interpretation

The calibration and the determinant in Eq. (11) play complementary roles. The calibration in Eq. (10) preserves the sample-level meaning of the score: when only one sample is selected, $J_{\mathrm{GFV}}(\{i\}) = s_i$. The determinant then supplies the batch-level correction. For larger sets, it rewards new Fisher-whitened directions and discounts directions already covered by the selected batch. GFV therefore turns independently high-scoring samples into a batch objective that favors complementary updates.

This volume-based objective also leads to an efficient selection rule. Under this nonnegative calibration, J_{GFV} is monotone submodular, so greedy maximization under $|S| \leq B$ attains the standard $(1 - 1/e)$ approximation guarantee. This property justifies the greedy batch-selection step used in Algorithm 1. Proposition 2 in Appendix D.1 gives the full structural statement and proof.

The construction can also be read in reverse: the log-determinant is not an arbitrary diversity correction but is singled out by the Fisher geometry.

Theorem 3 (Informal uniqueness of Gated Fisher Volume)**.** *Informally, Gated Fisher Volume is the unique Fisher-geometric batch extension of the sample score: it matches the gated sample-wise information score on singleton sets, is additive across independent Fisher subspaces, and obeys the same one-dimensional finite-information law.*

The formal statement and proof are deferred to the appendix D.2. The next remark gives its statistical meaning under the same local posterior approximation.

Remark 1 (Information-theoretic meaning of the batch objective)**.** *The log-determinant in* (11) *has a precise information-theoretic interpretation. Under the same local Gaussian and Gauss–Newton approximations used to derive the sample-level score, the realized batch information gain decomposes into three terms: a covariance-contraction term (the log-determinant), a non-positive trace correction, and a data-dependent mean-shift term. Taking the standard Bayesian-design expectation over the local observation eliminates the latter two, so that* $\frac{1}{2} J_{\mathrm{GFV}}(S)$ *equals the expected parameter-space information gain of the calibrated, alignment-gated batch experiment. The formal decomposition and proof are given in Appendix E.*

When the calibrated kernel in the high-dimensional representation is nearly diagonal, GFV becomes approximately additive and its redundancy correction weakens. We therefore evaluate batch redundancy in a low-dimensional representation of Fisher-whitened gradients (Appendix F.1). The projected kernel remains Fisher-derived and captures batch redundancy through a low-dimensional approximation, supported by projection diagnostics and downstream ablations (Appendices F.3 and F.4).

4.3 Algorithm of Batch Gated Fisher Volume

Exact maximization of (11) over all B-subsets is combinatorial. We use the greedy procedure implied by Proposition 2 in Appendix D.1: first form a shortlist R_t of the top-K candidates by sample score s_i ($K \geq B$; typically $K \in \{2B, 4B\}$), then run greedy GFV selection within this shortlist. The full procedure is given in Algorithm 1.

For $i, j \in R_t$, compute the Fisher inner product $C_{ij} = \boldsymbol{g}_i^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_j$ and calibrated kernel entry $K_{ij} = \sqrt{\alpha_i \alpha_j} C_{ij}$.

Given a partial selected set S, the Schur complement gives the marginal gain

$$\Delta_i(S) = \log\left(1 + \alpha_i I_i - \boldsymbol{k}_{iS}^\top (\boldsymbol{I} + \boldsymbol{K}_S)^{-1} \boldsymbol{k}_{Si}\right), \qquad i \in R_t \setminus S, \tag{12}$$

where $\boldsymbol{k}_{iS}$ is the vector of kernel entries between candidate i and the already selected set. When $S = \emptyset$, this reduces to

$$\Delta_i(\emptyset) = \log(1 + \alpha_i I_i) = s_i,$$

so the first greedy step coincides exactly with the original sample-level rule.

The diagonal Fisher implementation and the resulting computational cost are described in Appendix F.1.

Algorithm 1 Online Training with Batch Gated Fisher Volume

Input: model parameters $\boldsymbol{\theta}$, data D, and hyperparameters $(M, B, K, \eta, \lambda_F, \epsilon, \phi)$ with $K \geq B$

Initialize diagonal Fisher estimate $\boldsymbol{f}$
for each training iteration t **do**
  Draw candidate pool P_t and compute per-sample gradients $\{\boldsymbol{g}_i : i \in P_t\}$
  Sample scoring
  Set reference gradient $\bar{\boldsymbol{g}}_t \leftarrow |P_t|^{-1} \sum_{i \in P_t} \boldsymbol{g}_i$
  Compute I_i, A_i, $s_i = \phi(A_i) \log(1 + I_i)$, and α_i for all $i \in P_t$
  Form shortlist $R_t \leftarrow \mathrm{TopK}_{i \in P_t}(s_i)$
  Batch selection
  Build the calibrated Fisher kernel $K_{ij} = \sqrt{\alpha_i \alpha_j}\, \boldsymbol{g}_i^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_j$ on R_t
  Greedily select $S_t \subseteq R_t$ with $|S_t| = B$ using (12)
  Model update
  Update model on S_t: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} |S_t|^{-1} \sum_{i \in S_t} \ell(\boldsymbol{\theta}; z_i)$
  Update Fisher: $\boldsymbol{f} \leftarrow \lambda_F \boldsymbol{f} + (1 - \lambda_F) |P_t|^{-1} \sum_{i \in P_t} \boldsymbol{g}_i \odot \boldsymbol{g}_i$
end for

5 Experiments

5.1 Experimental Setup

Training data and models. We construct a $\sim$80K-example instruction-response corpus from five public sources—ALPACA [44], CODEALPACA [7], SHAREGPT[4], MATH [17], and GSM8K [11]—spanning instruction following, code generation, and reasoning, and apply LoRA [20] fine-tuning on LLAMA3.1-8B [15] and QWEN3-8B [49].

Baselines and benchmarks. We evaluate two variants of our method—**Ours-Gate** (Eq. (9)) and **Ours-GFV** (Eq. (11), Algorithm 1)—against Full training and six 10%-budget selection baselines: Random [47], GREATS [45], LEAD [29], UDS [52], GIST [37], and SPICE [6]. We report MMLU [16], GSM8K [11], and HumanEval [8]; further details are in Appendices F.1.

5.2 Main Results

Table 1 compares Ours-Gate and Ours-GFV with Full training and prior selection baselines under a fixed 10% selection budget. We report benchmark scores and their average across two models.

Ours-GFV gives the strongest overall performance. Ours-GFV achieves the best average score on both models, improving over Full training by $+2.29$ points on LLAMA3.1-8B and $+2.37$ points on QWEN3-8B. GFV is also the only method that outperforms Full training on all benchmarks across both models under the same budget.

The gate is competitive, and GFV provides a further lift. Ours-Gate performs competitively with prior methods, especially on LLAMA3.1-8B, where it exceeds every baseline average. Adding GFV further improves the average by $+0.76$ points on LLAMA3.1-8B and $+1.25$ points on QWEN3-8B. This supports the value of combining sample-level alignment with a batch-level diversity objective.

A 10% budget can match or exceed full-data training. Despite using 10% of the pool, Ours-GFV outperforms Full training on all six benchmark–model pairs. The largest gains appear on GSM8K for LLAMA3.1-8B ($+4.32$) and HumanEval for QWEN3-8B ($+4.88$), suggesting that selection can remove low-utility or conflicting updates rather than merely approximate full-data training.

5.3 Selection Geometry and Efficiency Analysis

Selection geometry in the (A_i, I_i) plane. To examine how the gate changes the selected subset, we visualize candidates in the alignment–information plane during training (Fig. 2(a)), which suggests

[4]The original ShareGPT sharing service (https://sharegpt.com/) is no longer maintained. In this work, we use a publicly curated ShareGPT-derived dataset hosted on Hugging Face: `anon8231489123/ShareGPT_Vicuna_unfiltered`.

Table 1: Main results under a fixed 10% selection budget. **Bold**: best result per row; underline: second best. Δ: gain of Ours-GFV over Full training.

Benchmark	Baselines							Ours		Δ
	Full	Random	GREATS	LEAD	UDS	GIST	SPICE	Gate	GFV	
llama3.1-8B										
GSM8K	44.73	43.06	46.10	38.78	40.49	40.71	39.20	46.55	**49.05**	+4.32
MMLU	62.54	62.74	59.82	63.29	62.98	**63.62**	61.84	63.48	63.27	+0.73
HumanEval	38.41	37.80	37.20	36.22	38.41	39.63	37.20	**40.24**	**40.24**	+1.83
Average	48.56	47.87	47.71	46.10	47.29	47.99	46.08	50.09	**50.85**	+2.29
Qwen3-8B										
GSM8K	84.76	83.40	82.87	84.38	85.06	**85.57**	83.17	83.70	85.52	+0.76
MMLU	72.62	73.75	71.55	73.52	73.81	73.88	73.43	73.99	**74.09**	+1.47
HumanEval	76.83	79.88	70.73	72.56	79.27	81.32	78.05	79.88	**81.71**	+4.88
Average	78.07	79.01	75.05	76.82	79.38	80.26	78.22	79.19	**80.44**	+2.37

an approximate four-region interpretation. Candidates with both high information and high alignment occupy the upper-right region and are consistently kept. High-information but weaker-alignment candidates appear toward the upper-left and are often dropped by the gate despite their large I_i. Lower-information but strongly aligned candidates appear toward the lower-right and are promoted because their updates are more coherent with the current descent direction. Candidates with limited information and limited alignment lie toward the lower-left and are rarely selected. This geometry shows that the gate turns information-only ranking into an alignment-aware criterion, suppressing poorly aligned high-information updates while recovering coherent lower-information updates.

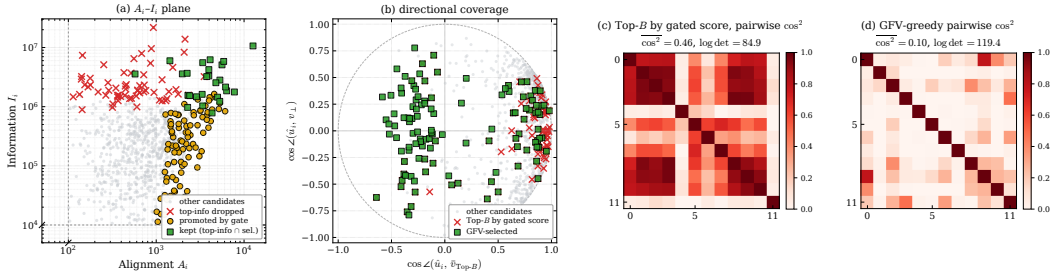


Figure 2: **Selection geometry induced by alignment gating and GFV. (a)** In the (A_i, I_i) plane, the gate drops high-I_i but low-A_i samples (red) and promotes well-aligned ones (orange), keeping the rest (green). **(b)** GFV spreads selected samples across complementary whitened-gradient directions. **(c) and (d)** Pairwise $\cos^2$ matrices show that GFV reduces redundancy among the selected Fisher-whitened directions.

GFV reshapes gated selection. We compare GFV with a baseline that selects the Top-B samples by the same gated sample score s_i. GFV computes batch redundancy in an 8-dimensional projected Fisher-whitened space. We measure redundancy by average pairwise squared cosine similarity between normalized projected directions. Fig. 2(b)–(d) shows that GFV changes the geometry of gated selection on LLAMA3.1-8B. The Top-B baseline concentrates selected samples near the dominant gated-score direction, whereas GFV spreads them toward complementary gradient directions. For the displayed mini-batch, GFV reduces $\overline{\cos^2}$ from 0.46 to 0.10 and raises J_{GFV} from 84.9 to 119.4 ($\Delta J_{\mathrm{GFV}} = +34.5$). Across the visualized mini-batches, GFV improves J_{GFV} at every step, with mean $\Delta J_{\mathrm{GFV}} = +8.41$. This confirms that the diversity term reshapes a high-score candidate set into a less redundant, higher-volume update batch.

Budget sensitivity. Fig. 3 shows that Ours-GFV is highly data-efficient. The average performance rises quickly as the selected subset expands and exceeds the full-data training reference once the budget reaches a moderate range. The best performance is obtained around the 10% budget used in our main experiments, while larger budgets remain competitive but provide limited additional gain. This trend suggests that GFV captures useful training signal with a compact subset, and that adding more data eventually introduces redundant or lower-utility examples.

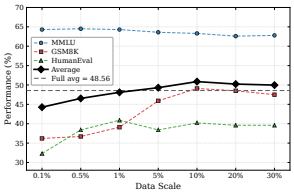


Figure 3: Budget sensitivity. Smaller budget is preferable at similar performance.

GFV improves performance with moderate training-time overhead. We also evaluate the wall-clock cost of each method on LLAMA3.1-8B and report training time in Fig. 4. Full training and Random selection are the fastest, as they avoid additional scoring or batch-level optimization. Among criterion-based selection methods, Ours-GFV has the lowest runtime overhead, requiring $1.56\times$ the time of Full training. Thus, the performance gains of GFV do not come from the largest computational budget; instead, GFV occupies a favorable point on the accuracy–efficiency frontier among the compared selection methods.

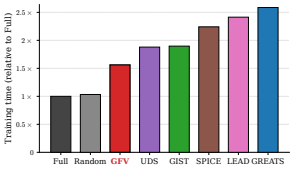


Figure 4: Runtime comparison. Lower is better.

5.4 Ablation Study

We ablate the two key components of our selector on LLAMA3.1-8B under the same 10% selection budget: the alignment gate $\phi(A_i)$ and the batch-level Gated Fisher Volume (GFV) objective. Table 2 reports all four combinations.

The gate and GFV provide complementary gains. Starting from the no-gating baseline with an average score of 49.72, adding only the alignment gate improves the average to 50.09, while applying GFV without the gate improves it to 50.30. Combining both components gives the best average score of 50.85, improving over the no-gating baseline by $+1.13$ points. This indicates that sample-level alignment and batch-level diversity contribute complementary benefits to data selection.

Table 2: Component ablation.

Variant	Modules		Benchmarks			Avg.
	Gate	GFV	MMLU	GSM8K	HE	
No gate			63.35	46.17	39.63	49.72
Gate	✓		63.48	46.55	40.24	50.09
GFV only		✓	62.90	47.10	40.90	50.30
Gate+GFV	✓	✓	**63.27**	**49.05**	**40.24**	**50.85**

6 Related Work

Online and dynamic data selection. Data selection methods can be broadly divided into offline subset construction and online re-evaluation during optimization. Offline methods select training data before optimization using distribution matching, curation, or gradient similarity, while online methods adapt sample value to the evolving model state [1, 31]. Existing online selectors use several forms of model-dependent signal. Loss-based and importance-sampling methods rely on loss or gradient norm [22, 23]. Reducible-loss and training-dynamics methods track how examples evolve during optimization [38]. Recent LLM-focused selectors further use Taylor-style utilities and utility–diversity criteria [45, 52]. These criteria make selection state-dependent, but primarily evaluate model response, training dynamics, or diversity rather than the induced parameter update.

Information- and geometry-based selectors are closest to our work: FisherSFT uses Fisher-based information gain [12], SPICE uses log-determinant Fisher information with conflict-aware penalties [6], and GIST uses gradient subspaces and projected alignment [37]. We share the update-centric view, and differ by deriving a local parameter-space information score at the current iterate, gating it by descent alignment, and lifting it to a calibrated Fisher-volume batch objective that preserves singleton scores while discounting redundant Fisher-whitened directions.

Information geometry and Bayesian data acquisition. Fisher geometry provides a local KL-based metric on predictive distributions. The Fisher–Rao metric gives a reparameterization-invariant

geometry for statistical models [39, 10], and natural gradient descent uses this geometry to precondition parameter updates [3, 35]. Scalable variants such as K-FAC extend this idea to neural networks [36], while related Fisher-geometric principles appear in trust-region policy optimization and natural-gradient variational inference [40, 19, 24]. These works use Fisher geometry to decide *how* to move parameters; we use it one step earlier, to decide *which data-induced movements are worth taking*. Bayesian experimental design selects observations by posterior utility, commonly through entropy reduction or KL information gain [33, 5]. Log-determinant and submodular criteria have been used to measure information volume in sensor placement and related acquisition problems [27, 18]. Our formulation connects these lines by treating each sample as a local Fisher experiment valued by its parameter-space information gain and its alignment with the current descent direction. In online fine-tuning, selected observations are training samples that immediately drive optimization, so the acquisition rule must also account for descent alignment and batch-level non-redundancy.

7 Conclusion

We presented *Alignment-Gated Fisher Volume*, a principled framework for online data selection grounded in Fisher information geometry. Starting from parameter-space information gain, we derive a Fisher-whitened information score under a local Gaussian approximation and gate it by first-order descent alignment, yielding a sample-level rule for informative, descent-aligned updates. At the batch level, we introduced a calibrated log-determinant objective that rewards complementary update directions. The theory shows that alignment is necessary for first-order descent consistency and that GFV has a monotone-submodular structure supporting greedy selection. Experiments under fixed supervised fine-tuning budgets show gains over representative online selectors, while ablations support the distinct roles of information, alignment, and batch diversity. This perspective casts data selection as choosing the updates most worth admitting into the current training step.

Acknowledgments and Disclosure of Funding

Yaping Wang acknowledges support from the National Natural Science Foundation of China (Grant Nos. 12271166, 12671308 and 12531013) and the National Key Research and Development Program of China (Grant No. 2024YFA1016200).

References

[1] Alon Albalak, Yanai Elazar, Sang Michael Xie, Shayne Longpre, Nathan Lambert, Xinyi Wang, Niklas Muennighoff, Bairu Hou, Liangming Pan, Haewon Jeong, et al. A survey on data selection for language models. *Transactions on Machine Learning Research*.

[2] Alen Alexanderian. Optimal experimental design for infinite-dimensional bayesian inverse problems governed by pdes: A review. *Inverse Problems*, 37(4):043001, 2021.

[3] Shun-Ichi Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

[4] Shun-ichi Amari. *Information geometry and its applications*. Springer, 2016.

[5] Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical science*, pages 273–304, 1995.

[6] Powei Chang, Jinpeng Zhang, Bowen Chen, Chenyu Wang, Chenlu Guo, Yixing Zhang, Yukang Gao, JianXiang Xiang, Yue Gao, Chaoqun Sun, Yiyi Chen, and Dongying kong. SPICE: Submodular penalized information–conflict selection for efficient large language model training. In *The Fourteenth International Conference on Learning Representations*, 2026.

[7] Sahil Chaudhary. Code alpaca: An instruction-following llama model for code generation. `https://github.com/sahil280114/codealpaca`, 2023.

[8] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

[9] Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. Training deep nets with sublinear memory cost, 2016.

[10] Nikolai Nikolaevich Chentsov. *Statistical decision rules and optimal inference*. American Mathematical Society, 1982.

[11] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[12] Rohan Deb, Kiran Koshy Thekumparampil, Kousha Kalantari, Gaurush Hiranandani, Shoham Sabach, and Branislav Kveton. Fishersft: Data-efficient supervised fine-tuning of language models using information gain. In *Forty-second International Conference on Machine Learning*.

[13] Ronald Aylmer Fisher. Theory of statistical estimation. In *Mathematical proceedings of the Cambridge philosophical society*, volume 22, pages 700–725. Cambridge University Press, 1925.

[14] Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. The language model evaluation harness, 07 2024.

[15] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[16] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

[17] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring Mathematical Problem Solving With the MATH Dataset. *arXiv e-prints*, page arXiv:2103.03874, March 2021.

[18] Philipp Hennig and Christian J Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(6), 2012.

[19] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of machine learning research*, 2013.

[20] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

[21] Zhongzhan Huang, Wenqi Shao, Xinjiang Wang, Liang Lin, and Ping Luo. Rethinking the pruning criteria for convolutional neural network. *Advances in Neural Information Processing Systems*, 34:16305–16318, 2021.

[22] Angela H Jiang, Daniel L-K Wong, Giulio Zhou, David G Andersen, Jeffrey Dean, Gregory R Ganger, Gauri Joshi, Michael Kaminksy, Michael Kozuch, Zachary C Lipton, et al. Accelerating deep learning by focusing on the biggest losers. *arXiv preprint arXiv:1910.00762*, 2019.

[23] Angelos Katharopoulos and François Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *International conference on machine learning*, pages 2525–2534. PMLR, 2018.

[24] Mohammad Khan and Wu Lin. Conjugate-computation variational inference: Converting variational inference in non-conjugate models to inferences in conjugate models. In *Artificial Intelligence and Statistics*, pages 878–887. PMLR, 2017.

[25] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[26] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR, 2017.

[27] Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9(2), 2008.

[28] Melvin Leok and Jun Zhang. Connecting information geometry and geometric mechanics. *Entropy*, 19(10):518, 2017.

[29] Xiaotian Lin, Yanlin Qi, Yizhang Zhu, Themis Palpanas, Chengliang Chai, Nan Tang, and Yuyu Luo. Lead: iterative data selection for efficient llm instruction tuning. *Proceedings of the VLDB Endowment*, 19(3):426–439, 2025.

[30] Liangxin Liu, Xuebo Liu, Derek F Wong, Dongfang Li, Ziyi Wang, Baotian Hu, and Min Zhang. Selectit: Selective instruction tuning for llms via uncertainty-aware self-reflection. *Advances in Neural Information Processing Systems*, 37:97800–97825, 2024.

[31] Ilya Loshchilov and Frank Hutter. Online batch selection for faster training of neural networks. *arXiv preprint arXiv:1511.06343*, 2015.

[32] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[33] David JC MacKay. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992.

[34] Jerrold E Marsden and Matthew West. Discrete mechanics and variational integrators. *Acta numerica*, 10:357–514, 2001.

[35] James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

[36] James Martens and Roger Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In *International conference on machine learning*, pages 2408–2417. PMLR, 2015.

[37] Guanghui Min, Tianhao Huang, Ke Wan, and Chen Chen. Gist: Targeted data selection for instruction tuning via coupled optimization geometry. *arXiv preprint arXiv:2602.18584*, 2026.

[38] Sören Mindermann, Jan M Brauner, Muhammed T Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, et al. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning*, pages 15630–15649. PMLR, 2022.

[39] C Radhakrishna Rao et al. Information and the accuracy attainable in the estimation of statistical parameters. *Bull. Calcutta Math. Soc*, 37(3):81–91, 1945.

[40] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015.

[41] Hirohiko Shima. *The geometry of Hessian structures*. World Scientific, 2007.

[42] Hwanjun Song, Minseok Kim, Sundong Kim, and Jae-Gil Lee. Carpe diem, seize the samples uncertain" at the moment" for adaptive batch selection. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1385–1394, 2020.

[43] Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A Smith, and Yejin Choi. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9275–9293, 2020.

[44] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

[45] Jiachen T. Wang, Tong Wu, Dawn Song, Prateek Mittal, and Ruoxi Jia. Greats: Online selection of high-quality data for llm training in every iteration. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 131197–131223. Curran Associates, Inc., 2024.

[46] Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: selecting influential data for targeted instruction tuning. In *Proceedings of the 41st International Conference on Machine Learning*, pages 54104–54132, 2024.

[47] Tingyu Xia, Bowen Yu, Kai Dang, An Yang, Yuan Wu, Yuan Tian, Yi Chang, and Junyang Lin. Rethinking data selection at scale: Random selection is almost all you need. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 2698–2711, Suzhou, China, November 2025. Association for Computational Linguistics.

[48] Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. Data selection for language models via importance resampling. *Advances in Neural Information Processing Systems*, 36:34201–34227, 2023.

[49] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[50] Bolin Zhang, Jiahao Wang, Qianlong Du, Jiajun Zhang, Zhiying Tu, and Dianhui Chu. A survey on data selection for llm instruction tuning. *Journal of Artificial Intelligence Research*, 83, 2025.

[51] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021, 2023.

[52] Heming Zou, Yixiu Mao, Yun Qu, Qi Wang, and Xiangyang Ji. Utility-diversity aware online batch selection for llm supervised fine-tuning. *arXiv preprint arXiv:2510.16882*, 2025.

A Additional Method Overview

Figure 5 summarizes the relationship between sample-level Fisher information, descent alignment, batch-level complementarity, GFV selection, and the corresponding theoretical properties.

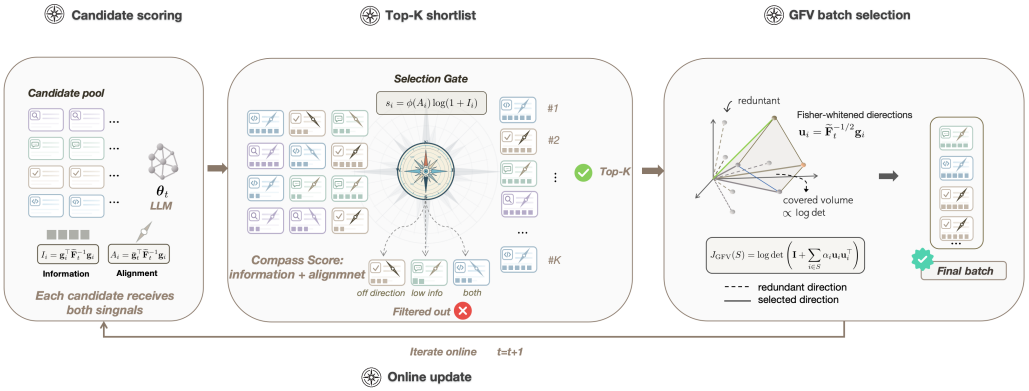


Figure 5: Additional overview of the proposed method.

B Proof of Theorem 1: Local Information-Gain Proxy

This section proves Theorem 1. We start from the local Gaussian posterior model in Section 3.2 and derive the exact rank-one formula that underlies the lightweight proxy used in the main text.

Proof of Theorem 1. Let

$$p_t(\boldsymbol{\theta}) \approx \mathcal{N}(\boldsymbol{\theta}_t, \boldsymbol{\Sigma}_t)$$

be the local posterior approximation at iteration t, and for sample i define

$$\boldsymbol{g}_i := \nabla_{\boldsymbol{\theta}} \ell_i(\boldsymbol{\theta}_t), \qquad \boldsymbol{H}_i := \nabla^2_{\boldsymbol{\theta}} \ell_i(\boldsymbol{\theta}_t).$$

Under the quadratic expansion of the sample loss around $\boldsymbol{\theta}_t$, the sample-updated local posterior remains Gaussian with

$$\boldsymbol{\Sigma}_{t+1}^{(i)} = (\boldsymbol{\Sigma}_t^{-1} + \boldsymbol{H}_i)^{-1}, \qquad \boldsymbol{\mu}_{t+1}^{(i)} = \boldsymbol{\theta}_t - (\boldsymbol{\Sigma}_t^{-1} + \boldsymbol{H}_i)^{-1} \boldsymbol{g}_i. \tag{13}$$

Therefore the realized local information gain

$$\mathrm{IG}_i := D_{\mathrm{KL}}\Big(p_{t+1}^{(i)}(\boldsymbol{\theta}) \,\|\, p_t(\boldsymbol{\theta})\Big)$$

is the KL divergence between two Gaussians. Using the standard Gaussian KL formula and substituting (13) yields

$$\mathrm{IG}_i = \frac{1}{2}\left[\log\left|\boldsymbol{I} + \boldsymbol{\Sigma}_t \boldsymbol{H}_i\right| - d + \mathrm{tr}\Big(\big(\boldsymbol{I} + \boldsymbol{\Sigma}_t^{1/2} \boldsymbol{H}_i \boldsymbol{\Sigma}_t^{1/2}\big)^{-1}\Big) + \boldsymbol{u}_i^\top (\boldsymbol{I} + \boldsymbol{B}_i)^{-2} \boldsymbol{u}_i\right], \tag{14}$$

where

$$\boldsymbol{B}_i := \boldsymbol{\Sigma}_t^{1/2} \boldsymbol{H}_i \boldsymbol{\Sigma}_t^{1/2}, \qquad \boldsymbol{u}_i := \boldsymbol{\Sigma}_t^{1/2} \boldsymbol{g}_i.$$

Now apply the Gauss–Newton reduction $\boldsymbol{H}_i \approx \boldsymbol{g}_i \boldsymbol{g}_i^\top$. Then

$$\boldsymbol{B}_i \approx \boldsymbol{u}_i \boldsymbol{u}_i^\top, \qquad r_i := \boldsymbol{g}_i^\top \boldsymbol{\Sigma}_t \boldsymbol{g}_i = \boldsymbol{u}_i^\top \boldsymbol{u}_i.$$

The matrix determinant lemma gives

$$\log|\boldsymbol{I} + \boldsymbol{B}_i| = \log(1 + r_i).$$

The Sherman–Morrison formula gives

$$(\boldsymbol{I} + \boldsymbol{B}_i)^{-1} = \boldsymbol{I} - \frac{\boldsymbol{u}_i \boldsymbol{u}_i^\top}{1 + r_i},$$

and therefore

$$\mathrm{tr}\big((\boldsymbol{I}+\boldsymbol{B}_i)^{-1}\big) = d - \frac{r_i}{1+r_i}.$$

Since $\boldsymbol{u}_i$ is an eigenvector of $\boldsymbol{B}_i$ with eigenvalue r_i,

$$(\boldsymbol{I}+\boldsymbol{B}_i)^{-2}\boldsymbol{u}_i = \frac{1}{(1+r_i)^2}\boldsymbol{u}_i,$$

so the mean-shift term becomes

$$\boldsymbol{u}_i^\top(\boldsymbol{I}+\boldsymbol{B}_i)^{-2}\boldsymbol{u}_i = \frac{r_i}{(1+r_i)^2}.$$

Substituting these identities into (14) gives the exact rank-one local formula

$$\mathrm{IG}_i = \frac{1}{2}\left[\log(1+r_i) - \frac{r_i^2}{(1+r_i)^2}\right]. \tag{15}$$

This expression immediately implies two facts.

First, IG_i is strictly increasing in r_i, because

$$\frac{d}{dr}\left(\log(1+r) - \frac{r^2}{(1+r)^2}\right) = \frac{1+r^2}{(1+r)^3} > 0 \qquad (r \geq 0).$$

Hence ranking by the exact local information gain is equivalent to ranking by r_i.

Second, the correction term has different interpretations in the small- and large-information regimes. For small r_i,

$$\frac{r_i^2}{(1+r_i)^2} = O(r_i^2) \qquad \text{as } r_i \to 0,$$

and therefore

$$\mathrm{IG}_i = \frac{1}{2}\log(1+r_i) + O(r_i^2). \tag{16}$$

For the high-information regime encountered in practice, $r_i \gg 1$, the same exact formula gives

$$\frac{r_i^2}{(1+r_i)^2} = 1 + O(r_i^{-1}),$$

so that

$$\mathrm{IG}_i = \frac{1}{2}\log(1+r_i) - \frac{1}{2} + O(r_i^{-1}). \tag{17}$$

Thus $\frac{1}{2}\log(1+r_i)$ serves as a lightweight finite-information surrogate: it agrees with the local expansion for small r_i, differs from the exact rank-one information gain by a bounded additive offset for large r_i, and preserves the same ordering because both IG_i and $\log(1+r_i)$ are strictly increasing functions of r_i.

Finally, under the Fisher identification $\boldsymbol{\Sigma}_t \approx \widetilde{\boldsymbol{F}}_t^{-1}$ introduced in Section 2.2, the quantity r_i becomes

$$r_i \approx \boldsymbol{g}_i^\top \widetilde{\boldsymbol{F}}_t^{-1}\boldsymbol{g}_i =: I_i,$$

which is exactly the practical information score used in the main text. □

C Alignment-Gated Sample Score

C.1 Derivation of the Gated Sample Score

Section 3.2 shows that the informative content of sample i is captured by the finite-information term

$$\psi(I_i) := \log(1+I_i), \qquad I_i = \boldsymbol{g}_i^\top \widetilde{\boldsymbol{F}}_t^{-1}\boldsymbol{g}_i.$$

On its own, however, this term measures only update magnitude in Fisher geometry. To obtain a descent-consistent rule, we must also control whether the induced update points along a direction that improves the current objective.

Consider the natural-gradient style single-sample update

$$\Delta\boldsymbol{\theta}_i = -\eta \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i,$$

and let $\boldsymbol{G}_t := \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_t)$ denote the objective gradient. A first-order Taylor expansion gives

$$\mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_i) - \mathcal{L}(\boldsymbol{\theta}_t) = -\eta\, \boldsymbol{G}_t^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i + O(\eta^2).$$

Therefore the oracle alignment statistic is

$$A_i^* := \boldsymbol{G}_t^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i.$$

If $A_i^* > 0$, the sample decreases the objective to first order; if $A_i^* < 0$, it increases the objective to first order. The ideal hard-gated rule is thus

$$s_i^{\mathrm{oracle}} := \psi(I_i)\, \mathbf{1}\{A_i^* > 0\}.$$

Since $\boldsymbol{G}_t$ is unavailable online, we replace it by a computable reference gradient $\bar{\boldsymbol{g}}_t$ and define the practical alignment statistic

$$A_i := \bar{\boldsymbol{g}}_t^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i.$$

Replacing the discontinuous indicator by a monotone nonnegative gate ϕ gives the smooth selection rule

$$s_i := \phi(A_i)\, \log(1 + I_i). \tag{18}$$

When $\phi(z) = \mathbf{1}\{z > 0\}$, (18) reduces to the hard gate; smooth choices such as a sigmoid retain the same alignment dependence while avoiding a discontinuous decision boundary.

C.2 Formal Necessity of Descent Alignment Gating

This subsection gives the formal result behind Theorem 2. The key point is that information alone can favor a batch whose first-order update opposes the objective, whereas a positively aligned replacement batch improves the first-order loss.

Proposition 1 (Formal necessity of descent alignment gating)**.** *Let*

$$I_i := \boldsymbol{g}_i^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i, \qquad A_i^* := \boldsymbol{G}_t^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i, \qquad \boldsymbol{G}_t := \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_t).$$

For any subset $S \subseteq \mathcal{P}_t$ with $|S| = B$, define the batch update

$$\Delta\boldsymbol{\theta}_S := -\frac{\eta}{B} \sum_{i \in S} \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i.$$

Assume that $\mathcal{L}$ is twice continuously differentiable in a neighborhood of $\boldsymbol{\theta}_t$, and that at least B candidates in $\mathcal{P}_t$ satisfy $A_i^* > 0$.
Let S_I be any batch selected by information alone, that is, any maximizer of $\sum_{i \in S} I_i$ over subsets of size B. If S_I contains at least one negatively aligned sample, then there exists a batch $S_+ \subseteq \mathcal{P}_t$ with $|S_+| = B$ and $A_i^* > 0$ for all $i \in S_+$ such that

$$\mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_+}) - \mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_I}) \le -\frac{\eta}{B} \sum_{i \in S_I :\, A_i^* < 0} |A_i^*| + C_t \eta^2 \tag{19}$$

for some constant $C_t > 0$ independent of η. In particular, if S_I contains at least one negatively aligned sample, then for all sufficiently small $\eta > 0$,

$$\mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_+}) < \mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_I}).$$

Thus pure information ranking can be strictly suboptimal already at first order, and dependence on descent alignment is necessary for descent-consistent online data selection.

Proof. Write

$$S_I^+ := \{i \in S_I : A_i^* > 0\}, \qquad S_I^- := \{i \in S_I : A_i^* < 0\}.$$

If S_I^- is nonempty, then $|S_I^+| < B$. Since at least B candidates in $\mathcal{P}_t$ satisfy $A_i^* > 0$, we may choose a set $R \subseteq \mathcal{P}_t \setminus S_I$ of size $|S_I^-|$ such that $A_i^* > 0$ for every $i \in R$. Define

$$S_+ := S_I^+ \cup R.$$

Then $|S_+| = B$ and every element of S_+ is positively aligned.

Because $\mathcal{L}$ is twice continuously differentiable, Taylor's theorem implies that there exists $C_t > 0$ such that for every batch S of size B,

$$\mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_S) = \mathcal{L}(\boldsymbol{\theta}_t) + \boldsymbol{G}_t^\top \Delta\boldsymbol{\theta}_S + R_t(S), \qquad |R_t(S)| \le C_t\eta^2$$

for all sufficiently small $\eta > 0$. Substituting the definition of $\Delta\boldsymbol{\theta}_S$ gives

$$\mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_S) = \mathcal{L}(\boldsymbol{\theta}_t) - \frac{\eta}{B}\sum_{i\in S} A_i^* + R_t(S).$$

Hence

$$\begin{aligned}&\mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_+}) - \mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_I}) \\ &\quad = -\frac{\eta}{B}\left(\sum_{i\in S_+} A_i^* - \sum_{i\in S_I} A_i^*\right) + R_t(S_+) - R_t(S_I).\end{aligned}$$

Now

$$\sum_{i\in S_+} A_i^* - \sum_{i\in S_I} A_i^* = \sum_{i\in R} A_i^* - \sum_{i\in S_I^-} A_i^*.$$

The first term on the right is nonnegative because every $i \in R$ has $A_i^* > 0$, while the second equals $\sum_{i\in S_I^-} |A_i^*|$. Therefore

$$\sum_{i\in S_+} A_i^* - \sum_{i\in S_I} A_i^* \ge \sum_{i\in S_I^-} |A_i^*|.$$

Using this inequality and the remainder bound gives

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_+}) - \mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_I}) &\le -\frac{\eta}{B}\sum_{i\in S_I^-} |A_i^*| + |R_t(S_+)| + |R_t(S_I)| \\ &\le -\frac{\eta}{B}\sum_{i\in S_I:\, A_i^*<0} |A_i^*| + 2C_t\eta^2.\end{aligned}$$

Absorbing the factor 2 into C_t proves (19).

If S_I^- is nonempty, then

$$\Gamma_t := \sum_{i\in S_I:\, A_i^*<0} |A_i^*| > 0.$$

Choosing $\eta > 0$ small enough that $C_t\eta < \Gamma_t/B$ yields the strict inequality

$$\mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_+}) < \mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_I}).$$

□

Lemma 1 (Deviation bound for the practical alignment proxy)**.** *Let*

$$A_i := \bar{\boldsymbol{g}}_t^\top \widetilde{\boldsymbol{F}}_t^{-1}\boldsymbol{g}_i, \qquad A_i^* := \boldsymbol{G}_t^\top \widetilde{\boldsymbol{F}}_t^{-1}\boldsymbol{g}_i.$$

Then, for every candidate i,

$$|A_i - A_i^*| \le \|\bar{\boldsymbol{g}}_t - \boldsymbol{G}_t\|_{\widetilde{\boldsymbol{F}}_t^{-1}}\sqrt{I_i}, \qquad \|\boldsymbol{v}\|_{\widetilde{\boldsymbol{F}}_t^{-1}} := \sqrt{\boldsymbol{v}^\top \widetilde{\boldsymbol{F}}_t^{-1}\boldsymbol{v}}. \tag{20}$$

Consequently, if

$$|A_i^*| > \|\bar{\boldsymbol{g}}_t - \boldsymbol{G}_t\|_{\widetilde{\boldsymbol{F}}_t^{-1}}\sqrt{I_i},$$

then

$$\operatorname{sign}(A_i) = \operatorname{sign}(A_i^*).$$

Proof. By definition,

$$A_i - A_i^* = (\bar{\boldsymbol{g}}_t - \boldsymbol{G}_t)^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i.$$

Write this as an inner product in Fisher-whitened coordinates:

$$A_i - A_i^* = \big(\widetilde{\boldsymbol{F}}_t^{-1/2}(\bar{\boldsymbol{g}}_t - \boldsymbol{G}_t)\big)^\top \big(\widetilde{\boldsymbol{F}}_t^{-1/2} \boldsymbol{g}_i\big).$$

Applying Cauchy–Schwarz yields

$$|A_i - A_i^*| \le \|\widetilde{\boldsymbol{F}}_t^{-1/2}(\bar{\boldsymbol{g}}_t - \boldsymbol{G}_t)\|_2 \cdot \|\widetilde{\boldsymbol{F}}_t^{-1/2} \boldsymbol{g}_i\|_2 = \|\bar{\boldsymbol{g}}_t - \boldsymbol{G}_t\|_{\widetilde{\boldsymbol{F}}_t^{-1}} \sqrt{I_i},$$

which proves (20). If the right-hand side is strictly smaller than $|A_i^*|$, then A_i cannot cross zero relative to A_i^*, so

$$\operatorname{sign}(A_i) = \operatorname{sign}(A_i^*).$$

This proves the sign consistency statement. □

Corollary 1 (Sign consistency of the practical gate)**.** *If*

$$|A_i^*| > \|\bar{\boldsymbol{g}}_t - \boldsymbol{G}_t\|_{\widetilde{\boldsymbol{F}}_t^{-1}} \sqrt{I_i} \qquad \textit{for every } i \in \mathcal{P}_t,$$

then

$$\{i \in \mathcal{P}_t : A_i > 0\} = \{i \in \mathcal{P}_t : A_i^* > 0\}.$$

Thus the practical gate and the oracle gate induce the same positive-alignment set.

Proof. The conclusion follows directly from Lemma 1: if the deviation is smaller than the oracle margin for every sample, then the sign of the practical statistic equals the sign of the oracle statistic pointwise. □

Corollary 2 (First-order dominance of the practical gate)**.** *Assume the conditions of Proposition 1. If, in addition,*

$$|A_i^*| > \|\bar{\boldsymbol{g}}_t - \boldsymbol{G}_t\|_{\widetilde{\boldsymbol{F}}_t^{-1}} \sqrt{I_i} \qquad \textit{for every } i \in \mathcal{P}_t,$$

then any batch S_A obtained by replacing the practically negative samples in S_I by practically positive candidates satisfies

$$\mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_A}) < \mathcal{L}(\boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_{S_I})$$

for all sufficiently small $\eta > 0$, whenever S_I contains at least one oracle-negatively aligned sample.

Proof. By Corollary 1, the practical positive set coincides with the oracle positive set. The replacement construction in the proof of Proposition 1 therefore applies verbatim to S_A. □

Taken together, Proposition 1, Lemma 1, and Corollaries 1–2 justify the gated score in (18): the hard gate is first-order necessary, and the smooth gate $\phi(A_i)$ is its stable practical relaxation.

D Batch Score: Structural Properties and Axiomatic Characterization

D.1 Structural Properties of Gated Fisher Volume

Proposition 2 (Structural properties of Gated Fisher Volume)**.** *Let*

$$C_{ij} := \boldsymbol{u}_i^\top \boldsymbol{u}_j = \boldsymbol{g}_i^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_j, \qquad (\boldsymbol{K}_S)_{ij} := \sqrt{\alpha_i \alpha_j}\, C_{ij}.$$

Then the objective in (11) *satisfies:*

1. ***Singleton consistency:*** *for every candidate i,*

$$J_{\mathrm{GFV}}(\{i\}) = \log(1 + \alpha_i I_i) = s_i = \phi(A_i) \log(1 + I_i). \tag{21}$$

2. ***Orthogonal additivity:*** *if $\boldsymbol{u}_i^\top \boldsymbol{u}_j = 0$ for all $i \neq j$ in S, then*

$$J_{\mathrm{GFV}}(S) = \sum_{i \in S} s_i. \tag{22}$$

3. ***Finite redundancy discount:*** *if*

$$\boldsymbol{D}_S := \mathrm{diag}(1 + \alpha_i I_i)_{i \in S}, \qquad \boldsymbol{R}_S := \boldsymbol{D}_S^{-1/2}(\boldsymbol{I} + \boldsymbol{K}_S)\boldsymbol{D}_S^{-1/2},$$

then

$$J_{\mathrm{GFV}}(S) = \sum_{i \in S} s_i + \log\det(\boldsymbol{R}_S) \leq \sum_{i \in S} s_i, \tag{23}$$

where the gap $-\log\det(\boldsymbol{R}_S) \geq 0$ measures redundancy among the selected Fisher-whitened directions.

4. ***Monotonicity and submodularity:*** *with $\boldsymbol{v}_i := \sqrt{\alpha_i}\boldsymbol{u}_i$ and $\boldsymbol{M}_S := \boldsymbol{I} + \sum_{j \in S} \boldsymbol{v}_j \boldsymbol{v}_j^\top$, the marginal gain satisfies*

$$\Delta_i(S) := J_{\mathrm{GFV}}(S \cup \{i\}) - J_{\mathrm{GFV}}(S) = \log\left(1 + \boldsymbol{v}_i^\top \boldsymbol{M}_S^{-1} \boldsymbol{v}_i\right) \geq 0, \tag{24}$$

and if $S \subseteq T$, then $\Delta_i(T) \leq \Delta_i(S)$.

Consequently, greedy maximization under $|S| \leq B$ attains the standard $(1 - 1/e)$ approximation guarantee.

Proof. Let

$$\boldsymbol{v}_i := \sqrt{\alpha_i}\,\boldsymbol{u}_i, \qquad \boldsymbol{V}_S := [\boldsymbol{v}_i]_{i \in S} \in \mathbb{R}^{d \times |S|}, \qquad \boldsymbol{K}_S := \boldsymbol{V}_S^\top \boldsymbol{V}_S.$$

Then

$$J_{\mathrm{GFV}}(S) = \log\det\left(\boldsymbol{I} + \sum_{i \in S} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right) = \log\det(\boldsymbol{I} + \boldsymbol{V}_S \boldsymbol{V}_S^\top).$$

By Sylvester's determinant identity,

$$J_{\mathrm{GFV}}(S) = \log\det(\boldsymbol{I} + \boldsymbol{K}_S). \tag{25}$$

1. Singleton consistency. If $S = \{i\}$, then $\boldsymbol{K}_S = [\alpha_i \boldsymbol{u}_i^\top \boldsymbol{u}_i] = [\alpha_i I_i]$, so

$$J_{\mathrm{GFV}}(\{i\}) = \log(1 + \alpha_i I_i).$$

Using the calibration

$$\alpha_i = \frac{e^{s_i} - 1}{I_i}$$

for $I_i > 0$ gives $\log(1 + \alpha_i I_i) = s_i = \phi(A_i)\log(1 + I_i)$. The case $I_i = 0$ follows by continuity.

2. Orthogonal additivity. If $\boldsymbol{u}_i^\top \boldsymbol{u}_j = 0$ for all $i \neq j$ in S, then $\boldsymbol{v}_i^\top \boldsymbol{v}_j = 0$ as well, hence $\boldsymbol{K}_S$ is diagonal with entries $\alpha_i I_i$. Therefore,

$$J_{\mathrm{GFV}}(S) = \log\det(\boldsymbol{I} + \boldsymbol{K}_S) = \sum_{i \in S} \log(1 + \alpha_i I_i) = \sum_{i \in S} s_i.$$

3. Finite redundancy discount. Let

$$\boldsymbol{D}_S := \mathrm{diag}(1 + \alpha_i I_i)_{i \in S}, \qquad \boldsymbol{R}_S := \boldsymbol{D}_S^{-1/2}(\boldsymbol{I} + \boldsymbol{K}_S)\boldsymbol{D}_S^{-1/2}.$$

Then

$$\boldsymbol{I} + \boldsymbol{K}_S = \boldsymbol{D}_S^{1/2} \boldsymbol{R}_S \boldsymbol{D}_S^{1/2},$$

so

$$J_{\mathrm{GFV}}(S) = \log\det(\boldsymbol{D}_S) + \log\det(\boldsymbol{R}_S) = \sum_{i \in S} \log(1 + \alpha_i I_i) + \log\det(\boldsymbol{R}_S).$$

By singleton consistency, this becomes

$$J_{\mathrm{GFV}}(S) = \sum_{i \in S} s_i + \log\det(\boldsymbol{R}_S).$$

Moreover, $\boldsymbol{I} + \boldsymbol{K}_S \succeq 0$ and $\boldsymbol{D}_S \succ 0$, hence $\boldsymbol{R}_S \succeq 0$. By construction $\boldsymbol{R}_S$ has unit diagonal. Hadamard's inequality therefore gives

$$\det(\boldsymbol{R}_S) \leq 1,$$

with equality exactly when the normalized directions are mutually orthogonal. Hence

$$\log\det(\boldsymbol{R}_S) \leq 0,$$

which proves the redundancy discount.

4. Monotonicity and submodularity. Define

$$\boldsymbol{M}_S := \boldsymbol{I} + \sum_{j \in S} \boldsymbol{v}_j \boldsymbol{v}_j^\top.$$

Then for $i \notin S$,

$$J_{\mathrm{GFV}}(S \cup \{i\}) = \log\det(\boldsymbol{M}_S + \boldsymbol{v}_i \boldsymbol{v}_i^\top).$$

Applying the matrix determinant lemma yields

$$\Delta_i(S) := J_{\mathrm{GFV}}(S \cup \{i\}) - J_{\mathrm{GFV}}(S) = \log\left(1 + \boldsymbol{v}_i^\top \boldsymbol{M}_S^{-1} \boldsymbol{v}_i\right) \geq 0. \tag{26}$$

This proves monotonicity. If $S \subseteq T$, then $\boldsymbol{M}_T \succeq \boldsymbol{M}_S \succ 0$, hence $\boldsymbol{M}_T^{-1} \preceq \boldsymbol{M}_S^{-1}$. Therefore

$$\boldsymbol{v}_i^\top \boldsymbol{M}_T^{-1} \boldsymbol{v}_i \leq \boldsymbol{v}_i^\top \boldsymbol{M}_S^{-1} \boldsymbol{v}_i,$$

and since $\log(1+x)$ is increasing,

$$\Delta_i(T) \leq \Delta_i(S).$$

Thus J_{GFV} is submodular. The standard $(1 - 1/e)$ greedy guarantee under a cardinality constraint follows from monotone submodularity. □

This proves the full structural properties, including the claims highlighted in Proposition 2.

D.2 Full Axiomatic Characterization

This subsection gives the full theorem behind the informal uniqueness statement in Section 4.2. The claim is a conditional uniqueness statement: once the batch valuation is required to respect Fisher-whitened geometry, additivity across independent subspaces, and the one-dimensional finite-information law, the log-determinant form is forced.

Theorem 4 (Axiomatic characterization of Gated Fisher Volume)**.** *Let*

$$\boldsymbol{u}_i := \widetilde{\boldsymbol{F}}_t^{-1/2} \boldsymbol{g}_i, \qquad I_i := \|\boldsymbol{u}_i\|_2^2, \qquad s_i := \phi(A_i)\log(1 + I_i).$$

Consider any batch valuation of the form

$$J(S) = \Psi\left(\sum_{i \in S} \alpha_i \boldsymbol{u}_i \boldsymbol{u}_i^\top\right),$$

where $\alpha_i \geq 0$ and Ψ is a continuous functional on positive semidefinite matrices satisfying:

1. ***Normalization:*** $\Psi(0) = 0$.
2. ***Orthogonal invariance:*** $\Psi(\boldsymbol{Q}\boldsymbol{M}\boldsymbol{Q}^\top) = \Psi(\boldsymbol{M})$ *for every orthogonal matrix* $\boldsymbol{Q}$.
3. ***Additivity on independent subspaces:*** *if* $\boldsymbol{M}, \boldsymbol{N} \succeq 0$ *and* $\boldsymbol{M}\boldsymbol{N} = 0$*, then* $\Psi(\boldsymbol{M} + \boldsymbol{N}) = \Psi(\boldsymbol{M}) + \Psi(\boldsymbol{N})$.
4. ***One-dimensional finite-information law:*** *for every unit vector* $\boldsymbol{e}$ *and every* $x \geq 0$*,*

$$\Psi(x\boldsymbol{e}\boldsymbol{e}^\top) = \log(1 + x).$$

5. ***Singleton consistency:*** *$J(\{i\}) = s_i$ for every sample i.*

Then, for every i with $I_i > 0$,

$$\alpha_i = \frac{e^{s_i} - 1}{I_i} = \frac{(1 + I_i)^{\phi(A_i)} - 1}{I_i}. \tag{27}$$

and for $I_i = 0$ we adopt the continuous convention $\alpha_i = \phi(A_i)$. Moreover, for every batch S,

$$J(S) = \log\det\left(\boldsymbol{I} + \sum_{i \in S} \alpha_i \boldsymbol{u}_i \boldsymbol{u}_i^\top\right). \tag{28}$$

Therefore the calibrated Gated Fisher Volume objective is the unique Fisher-geometric batch extension of the sample score under this continuity convention.

Proof. Let $\boldsymbol{M} \succeq 0$ be any positive semidefinite matrix with spectral decomposition

$$\boldsymbol{M} = \sum_{k=1}^{r} \lambda_k \boldsymbol{q}_k \boldsymbol{q}_k^\top, \qquad \lambda_k \geq 0,$$

where the $\boldsymbol{q}_k$ are orthonormal. By orthogonal invariance, the value of $\Psi(\lambda \boldsymbol{q}\boldsymbol{q}^\top)$ depends only on the scalar λ and not on the particular unit vector $\boldsymbol{q}$. Indeed, if $\boldsymbol{Q}\boldsymbol{e} = \boldsymbol{q}$ for some orthogonal $\boldsymbol{Q}$, then

$$\Psi(\lambda \boldsymbol{q}\boldsymbol{q}^\top) = \Psi\big(\boldsymbol{Q}(\lambda \boldsymbol{e}\boldsymbol{e}^\top)\boldsymbol{Q}^\top\big) = \Psi(\lambda \boldsymbol{e}\boldsymbol{e}^\top) = \log(1 + \lambda)$$

by the one-dimensional finite-information law.

Now define the rank-one spectral components

$$\boldsymbol{M}_k := \lambda_k \boldsymbol{q}_k \boldsymbol{q}_k^\top.$$

For $k \neq \ell$, we have $\boldsymbol{M}_k \boldsymbol{M}_\ell = 0$ because $\boldsymbol{q}_k^\top \boldsymbol{q}_\ell = 0$. Repeatedly applying additivity on independent subspaces gives

$$\Psi(\boldsymbol{M}) = \sum_{k=1}^{r} \Psi(\boldsymbol{M}_k) = \sum_{k=1}^{r} \log(1 + \lambda_k).$$

Since the eigenvalues of $\boldsymbol{I} + \boldsymbol{M}$ are $1 + \lambda_1, \ldots, 1 + \lambda_r$ together with additional ones equal to 1, we obtain

$$\Psi(\boldsymbol{M}) = \log \prod_{k=1}^{r} (1 + \lambda_k) = \log\det(\boldsymbol{I} + \boldsymbol{M}).$$

Thus every admissible functional Ψ must equal $\boldsymbol{M} \mapsto \log\det(\boldsymbol{I} + \boldsymbol{M})$ on the cone of positive semidefinite matrices.

Applying this to

$$\boldsymbol{M}_S := \sum_{i \in S} \alpha_i \boldsymbol{u}_i \boldsymbol{u}_i^\top$$

gives

$$J(S) = \Psi(\boldsymbol{M}_S) = \log\det(\boldsymbol{I} + \boldsymbol{M}_S),$$

which is exactly (28).

It remains to determine the calibration coefficients. For a singleton set $\{i\}$,

$$s_i = J(\{i\}) = \log\det(\boldsymbol{I} + \alpha_i \boldsymbol{u}_i \boldsymbol{u}_i^\top) = \log(1 + \alpha_i \boldsymbol{u}_i^\top \boldsymbol{u}_i) = \log(1 + \alpha_i I_i),$$

where the last step again uses the rank-one determinant formula. Solving for α_i gives (27) whenever $I_i > 0$.

If $I_i = 0$, then $\boldsymbol{u}_i = 0$ and singleton consistency reduces to $s_i = \log(1) = 0$, which is automatically satisfied because $s_i = \phi(A_i)\log(1 + I_i) = 0$. In this degenerate case singleton consistency does not

uniquely determine α_i; the natural continuous extension is obtained from

$$\frac{(1+I)^{\phi(A_i)}-1}{I} \to \phi(A_i) \qquad \text{as } I \to 0,$$

so one sets $\alpha_i = \phi(A_i)$ by continuity.

This proves both uniqueness and the explicit GFV form. □

E Gated Fisher Volume as Expected Information Gain

This section shows that the log-determinant batch objective is not an ad-hoc diversity regularizer: under the same local Gaussian and Gauss–Newton approximations used in Sections 3–4, it equals the expected parameter-space information gain of the alignment-gated batch experiment.

We use the notation from Section 4 throughout: $\boldsymbol{u}_i = \widetilde{\boldsymbol{F}}_t^{-1/2}\boldsymbol{g}_i$, $I_i = \|\boldsymbol{u}_i\|^2$, $\alpha_i = (e^{s_i}-1)/I_i$, $\boldsymbol{M}_S = \boldsymbol{I} + \sum_{i\in S}\alpha_i \boldsymbol{u}_i\boldsymbol{u}_i^\top$, $J_{\mathrm{GFV}}(S) = \log\det \boldsymbol{M}_S$, and $\boldsymbol{q}_S = \sum_{i\in S}\alpha_i\boldsymbol{u}_i$.

E.1 Realized Batch KL Decomposition

Proposition 3 (Realized KL decomposition)**.** *Assume a local Gaussian posterior*

$$p_t(\boldsymbol{\theta}) \approx \mathcal{N}(\boldsymbol{\theta}_t, \boldsymbol{\Sigma}_t)$$

and approximate each selected sample by the local Gauss–Newton quadratic

$$\ell_i(\boldsymbol{\theta}) \approx \ell_i(\boldsymbol{\theta}_t) + \boldsymbol{g}_i^\top(\boldsymbol{\theta}-\boldsymbol{\theta}_t) + \frac{1}{2}(\boldsymbol{\theta}-\boldsymbol{\theta}_t)^\top \boldsymbol{g}_i\boldsymbol{g}_i^\top(\boldsymbol{\theta}-\boldsymbol{\theta}_t).$$

Consider the α-tempered batch likelihood obtained by replacing ℓ_i with $\alpha_i\ell_i$. Then the resulting local posterior is Gaussian, and its realized information gain

$$\mathcal{K}_\alpha(S) := D_{\mathrm{KL}}\big(p_{t,S}^{\alpha}(\boldsymbol{\theta}) \,\|\, p_t(\boldsymbol{\theta})\big)$$

is exactly

$$\mathcal{K}_\alpha(S) = \frac{1}{2}\left[\log\det \boldsymbol{M}_S - d + \mathrm{tr}(\boldsymbol{M}_S^{-1}) + \boldsymbol{q}_S^\top \boldsymbol{M}_S^{-2}\boldsymbol{q}_S\right], \qquad \boldsymbol{q}_S := \sum_{i\in S}\alpha_i\boldsymbol{u}_i.$$

In particular, the Gated Fisher Volume term $J_{\mathrm{GFV}}(S)$ is the log-determinant covariance contraction term in the realized batch KL.

Proof. The α-tempered batch posterior incorporates each sample likelihood raised to the power α_i. Under the Gauss–Newton approximation $\boldsymbol{H}_i \approx \boldsymbol{g}_i\boldsymbol{g}_i^\top$, the tempered negative log-likelihood contribution of sample i is

$$\alpha_i\ell_i(\boldsymbol{\theta}) \approx \mathrm{const} + \alpha_i\boldsymbol{g}_i^\top(\boldsymbol{\theta}-\boldsymbol{\theta}_t) + \frac{\alpha_i}{2}(\boldsymbol{\theta}-\boldsymbol{\theta}_t)^\top \boldsymbol{g}_i\boldsymbol{g}_i^\top(\boldsymbol{\theta}-\boldsymbol{\theta}_t).$$

Combining all $i \in S$ with the Gaussian prior $p_t(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}_t, \boldsymbol{\Sigma}_t)$, the tempered posterior remains Gaussian with precision and mean

$$\boldsymbol{\Sigma}_{S,\alpha}^{-1} = \boldsymbol{\Sigma}_t^{-1} + \sum_{i\in S}\alpha_i\boldsymbol{g}_i\boldsymbol{g}_i^\top = \widetilde{\boldsymbol{F}}_t + \sum_{i\in S}\alpha_i\boldsymbol{g}_i\boldsymbol{g}_i^\top,$$

$$\boldsymbol{\mu}_{S,\alpha} = \boldsymbol{\theta}_t - \boldsymbol{\Sigma}_{S,\alpha}\sum_{i\in S}\alpha_i\boldsymbol{g}_i.$$

We now whiten the coordinate system by the change of variable $\tilde{\boldsymbol{\theta}} = \widetilde{\boldsymbol{F}}_t^{1/2}(\boldsymbol{\theta}-\boldsymbol{\theta}_t)$. Under this transformation:

- the prior becomes $\tilde{\boldsymbol{\theta}} \sim \mathcal{N}(0, \boldsymbol{I}_d)$;

- the tempered posterior becomes $\tilde{\boldsymbol{\theta}} \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}_S, \tilde{\boldsymbol{\Sigma}}_S)$ with

$$\tilde{\boldsymbol{\Sigma}}_S^{-1} = \boldsymbol{I} + \sum_{i \in S} \alpha_i \, (\widetilde{\boldsymbol{F}}_t^{-1/2} \boldsymbol{g}_i)(\widetilde{\boldsymbol{F}}_t^{-1/2} \boldsymbol{g}_i)^\top = \boldsymbol{I} + \sum_{i \in S} \alpha_i \boldsymbol{u}_i \boldsymbol{u}_i^\top = \boldsymbol{M}_S,$$

  so $\tilde{\boldsymbol{\Sigma}}_S = \boldsymbol{M}_S^{-1}$, and

$$\tilde{\boldsymbol{\mu}}_S = -\boldsymbol{M}_S^{-1} \sum_{i \in S} \alpha_i \boldsymbol{u}_i = -\boldsymbol{M}_S^{-1} \boldsymbol{q}_S.$$

The KL divergence between two Gaussians $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and $\mathcal{N}(0, \boldsymbol{I}_d)$ in $\mathbb{R}^d$ is

$$D_{\mathrm{KL}} = \frac{1}{2}\big[-\log\det \boldsymbol{\Sigma} - d + \mathrm{tr}(\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\top \boldsymbol{\mu}\big].$$

Substituting $\boldsymbol{\Sigma} = \boldsymbol{M}_S^{-1}$ and $\boldsymbol{\mu} = -\boldsymbol{M}_S^{-1}\boldsymbol{q}_S$:

$$\begin{aligned}\mathcal{K}_\alpha(S) &= \frac{1}{2}\big[\log\det \boldsymbol{M}_S - d + \mathrm{tr}(\boldsymbol{M}_S^{-1}) + (\boldsymbol{M}_S^{-1}\boldsymbol{q}_S)^\top (\boldsymbol{M}_S^{-1}\boldsymbol{q}_S)\big] \\ &= \frac{1}{2}\big[\log\det \boldsymbol{M}_S - d + \mathrm{tr}(\boldsymbol{M}_S^{-1}) + \boldsymbol{q}_S^\top \boldsymbol{M}_S^{-2}\boldsymbol{q}_S\big].\end{aligned}$$

□

Proposition 3 decomposes the realized batch information gain into three terms with distinct roles:

- **Covariance contraction** ($\log\det \boldsymbol{M}_S$). This measures how much the batch narrows the posterior ellipsoid, and equals $J_{\mathrm{GFV}}(S)$.
- **Trace correction** ($\mathrm{tr}(\boldsymbol{M}_S^{-1}) - d$). Since $\boldsymbol{M}_S \succeq \boldsymbol{I}$, every eigenvalue of $\boldsymbol{M}_S$ is at least 1, so every eigenvalue of $\boldsymbol{M}_S^{-1}$ is at most 1. Hence $\mathrm{tr}(\boldsymbol{M}_S^{-1}) \le d$ and this term is always non-positive.
- **Mean-shift** ($\boldsymbol{q}_S^\top \boldsymbol{M}_S^{-2}\boldsymbol{q}_S$). This depends on the realized gradient directions and magnitudes. It is always non-negative, but is the source of noise sensitivity: it is large precisely when many selected gradients point in the same direction, which is the regime where outliers and mislabeled samples dominate.

Our batch objective retains only the covariance-contraction component. This is a deliberate design choice: the mean-shift term rewards batches whose aggregate gradient is large in Fisher-whitened coordinates, which can be dominated by a single outlier gradient. The log-determinant, by contrast, rewards batches that cover many independent Fisher directions.

E.2 GFV as Expected Information Gain

The log-determinant becomes the full KL objective after taking the standard Bayesian-design expectation over the local observation.

Proposition 4 (GFV as expected realized information gain)**.** *Consider the corresponding whitened local linear-Gaussian experiment*

$$\boldsymbol{\xi} \sim \mathcal{N}(0, \boldsymbol{I}), \qquad Y_i = (\sqrt{\alpha_i}\boldsymbol{u}_i)^\top \boldsymbol{\xi} + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0,1),$$

independently over i. Let $Y_S = \{Y_i : i \in S\}$. Then

$$\mathbb{E}_{Y_S}\left[D_{\mathrm{KL}}\big(p(\boldsymbol{\xi} \mid Y_S) \,\|\, p(\boldsymbol{\xi})\big)\right] = \frac{1}{2}\log\det\left(\boldsymbol{I} + \sum_{i \in S} \alpha_i \boldsymbol{u}_i \boldsymbol{u}_i^\top\right) = \frac{1}{2} J_{\mathrm{GFV}}(S).$$

Consequently, maximizing $J_{\mathrm{GFV}}(S)$ over $|S| = B$ is exactly equivalent to maximizing the expected realized KL of the calibrated gated local experiment.

Proof. In the linear-Gaussian experiment, the prior on $\boldsymbol{\xi}$ is $\mathcal{N}(0, \boldsymbol{I}_d)$ and the observation model is $Y_i = \sqrt{\alpha_i}\,\boldsymbol{u}_i^\top \boldsymbol{\xi} + \varepsilon_i$ with $\varepsilon_i \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$. Stacking the observations into a vector $\boldsymbol{y} \in \mathbb{R}^{|S|}$ with design

matrix $\boldsymbol{A} \in \mathbb{R}^{|S|\times d}$ whose i-th row is $\sqrt{\alpha_i}\,\boldsymbol{u}_i^\top$, the standard Bayesian linear regression posterior is

$$p(\boldsymbol{\xi} \mid Y_S) = \mathcal{N}\big(\boldsymbol{m}_S,\ (\boldsymbol{I} + \boldsymbol{A}^\top \boldsymbol{A})^{-1}\big), \qquad \boldsymbol{m}_S = (\boldsymbol{I} + \boldsymbol{A}^\top \boldsymbol{A})^{-1} \boldsymbol{A}^\top \boldsymbol{y}.$$

Since $\boldsymbol{A}^\top \boldsymbol{A} = \sum_{i\in S} \alpha_i \boldsymbol{u}_i \boldsymbol{u}_i^\top$, the posterior covariance is $(\boldsymbol{I} + \sum \alpha_i \boldsymbol{u}_i \boldsymbol{u}_i^\top)^{-1} = \boldsymbol{M}_S^{-1}$, which is data-independent. The posterior mean $\boldsymbol{m}_S$ is random through $\boldsymbol{y}$.

Writing the realized KL between $\mathcal{N}(\boldsymbol{m}_S, \boldsymbol{M}_S^{-1})$ and $\mathcal{N}(0, \boldsymbol{I}_d)$:

$$D_{\mathrm{KL}}\big(p(\boldsymbol{\xi} \mid Y_S) \,\|\, p(\boldsymbol{\xi})\big) = \frac{1}{2}\big[\log\det \boldsymbol{M}_S - d + \mathrm{tr}(\boldsymbol{M}_S^{-1}) + \|\boldsymbol{m}_S\|^2\big].$$

The first three terms are deterministic. For the random term, we compute $\mathrm{Cov}(\boldsymbol{m}_S)$. Since $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{\xi} + \boldsymbol{\varepsilon}$ with $\boldsymbol{\xi} \sim \mathcal{N}(0, \boldsymbol{I})$ and $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \boldsymbol{I}_{|S|})$ independent:

$$\begin{aligned}\mathrm{Cov}(\boldsymbol{m}_S) &= \boldsymbol{M}_S^{-1} \boldsymbol{A}^\top \mathrm{Cov}(\boldsymbol{y}) \boldsymbol{A}\, \boldsymbol{M}_S^{-1} \\ &= \boldsymbol{M}_S^{-1} \boldsymbol{A}^\top (\boldsymbol{A}\boldsymbol{A}^\top + \boldsymbol{I}_{|S|}) \boldsymbol{A}\, \boldsymbol{M}_S^{-1}.\end{aligned}$$

Using the identity $\boldsymbol{A}^\top(\boldsymbol{A}\boldsymbol{A}^\top + \boldsymbol{I})\boldsymbol{A} = \boldsymbol{A}^\top\boldsymbol{A}(\boldsymbol{A}^\top\boldsymbol{A} + \boldsymbol{I}) = (\boldsymbol{M}_S - \boldsymbol{I})\boldsymbol{M}_S$, this simplifies to

$$\mathrm{Cov}(\boldsymbol{m}_S) = \boldsymbol{M}_S^{-1}(\boldsymbol{M}_S - \boldsymbol{I})\boldsymbol{M}_S\, \boldsymbol{M}_S^{-1} = \boldsymbol{I} - \boldsymbol{M}_S^{-1}.$$

Therefore

$$\mathbb{E}[\|\boldsymbol{m}_S\|^2] = \mathrm{tr}\big(\mathrm{Cov}(\boldsymbol{m}_S)\big) = \mathrm{tr}(\boldsymbol{I} - \boldsymbol{M}_S^{-1}) = d - \mathrm{tr}(\boldsymbol{M}_S^{-1}).$$

Taking expectations of the realized KL:

$$\begin{aligned}\mathbb{E}\big[D_{\mathrm{KL}}(p(\boldsymbol{\xi} \mid Y_S) \,\|\, p(\boldsymbol{\xi}))\big] &= \frac{1}{2}\big[\log\det \boldsymbol{M}_S - d + \mathrm{tr}(\boldsymbol{M}_S^{-1}) + d - \mathrm{tr}(\boldsymbol{M}_S^{-1})\big] \\ &= \frac{1}{2}\log\det \boldsymbol{M}_S = \frac{1}{2} J_{\mathrm{GFV}}(S).\end{aligned}$$

The trace correction and expected mean-shift cancel exactly. □

Interpretation. The expectation in Proposition 4 is over the randomness in the observations Y_S, which corresponds to integrating out the data-dependent "surprise" before choosing which experiment to run. This is the standard Bayesian optimal experimental design criterion: among all size-B subsets, choose the one whose expected posterior contraction is largest. The cancellation is exact, not asymptotic, because the linear-Gaussian model has the special property that posterior covariance is data-independent while the posterior mean absorbs all observation randomness.

Together, Propositions 3 and 4 close the loop between the sample-level KL principle of Section 3 and the batch-level log-determinant of Section 4: the GFV objective is the expected information gain of the calibrated, alignment-gated local experiment, not an independent diversity heuristic.

F Additional Experiments

F.1 Implementation Details

Training data. Both models are fine-tuned on the same instruction-tuning pool of 78,976 examples. The pool is constructed from five sources: all 25,987 examples from Alpaca [44], all 20,016 examples from Alpaca-Code [7], a random sample of 18,000 examples from ShareGPT, all 7,500 examples from MATH [17], and all 7,473 training examples from GSM8K [11]. The resulting mixture contains 32.9% general instruction data, 25.3% code instruction data, 22.8% multi-turn dialogue data, 9.5% competition-style math data, and 9.5% grade-school math reasoning data.

Fine-tuning setup. All models are fine-tuned with LoRA [20] on the attention projection layers, using rank 16, scaling factor 32, and dropout 0.05 for all methods. Each method is trained for two epochs over its assigned training data: the *Full* baseline uses the entire pool, while all 10%-budget selection methods use their selected subset. Optimization uses AdamW [32] with bfloat16 training and gradient checkpointing [9].

Baseline implementation. All baseline methods are implemented with their official code releases. To isolate the effect of the selection criterion, we standardize the downstream fine-tuning configuration across all methods. Specifically, each method is run on the same LoRA configuration, optimizer, number of training epochs, maximum sequence length, and evaluation protocol.

Scope of related baselines. We do not exhaustively evaluate every method discussed in related work; instead, our main baselines cover representative and recent methods from the major relevant lines, including SPICE [6] for Fisher-information-based selection related to FisherSFT [12], and GIST [37] for gradient-geometry-based selection related to LESS [46].

Selection details. For our online selection methods, we select 12 candidates from each pool of 120 candidates. Ours-GFV uses a two-stage selection procedure: it first shortlists the top 48 candidates according to the gated Fisher-information score, and then selects 12 examples from this shortlist using the greedy GFV batch objective. Before scoring begins, we use 1,000 examples to warm up the diagonal Fisher estimate, so that early selection decisions are not made with an uninformative preconditioner. To preserve exploration freedom in the early stage of fine-tuning, the alignment gate is disabled during the first 100 optimization steps and is activated only afterwards. In implementation, we apply a Fisher-cosine normalization to the alignment statistic to reduce scale sensitivity. Candidate scores are computed using sequences truncated to 512 tokens, while the selected examples are subsequently trained with the full fine-tuning length of 1024 tokens. Because computing per-sample LoRA gradients for an entire candidate pool can exceed GPU memory, we compute the scores in small gradient micro-batches and then aggregate the scores over the full pool.

Diagonal Fisher computation. We implement GFV with a diagonal damped Fisher estimate

$$\widetilde{\boldsymbol{F}}_t \approx \mathrm{diag}(f_{1,t}+\epsilon, \ldots, f_{d,t}+\epsilon).$$

Under this approximation, the quantities used in the sample score are

$$I_i = \sum_{m=1}^{d} \frac{g_{i,m}^2}{f_{m,t}+\epsilon}, \qquad A_i = \sum_{m=1}^{d} \frac{\bar{g}_{t,m} g_{i,m}}{f_{m,t}+\epsilon}. \tag{29}$$

For batch selection, the Fisher inner product and calibrated kernel entry are

$$C_{ij} = \boldsymbol{g}_i^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_j = \sum_{m=1}^{d} \frac{g_{i,m} g_{j,m}}{f_{m,t}+\epsilon}, \qquad K_{ij} = \sqrt{\alpha_i \alpha_j}\, C_{ij}. \tag{30}$$

Computing all I_i and A_i over a candidate pool costs $O(Md)$ after gradients are available. The redundancy correction is evaluated only on the shortlist R_t, requiring $O(K^2 d)$ Fisher inner products. Greedy selection then operates on the resulting $K \times K$ kernel matrix; with Cholesky updates, the extra cost remains small when K is a modest multiple of B.

GFV gradient features. The greedy GFV rule needs the pairwise Fisher-whitened kernel $C_{ij} = \boldsymbol{u}_i^\top \boldsymbol{u}_j$, where $\boldsymbol{u}_i = \widetilde{\boldsymbol{F}}_t^{-1/2} \boldsymbol{g}_i \in \mathbb{R}^d$ and $d \sim 10^7$. In our high-dimensional representation, weak interactions between Fisher-whitened directions empirically make the GFV objective nearly additive, weakening batch-level redundancy correction. We therefore evaluate pairwise redundancy in a low-dimensional batch-local subspace. We obtain $\boldsymbol{z}_i$ in two steps: a fixed Gaussian random projection $\boldsymbol{G} \in \mathbb{R}^{m \times d}$ with $G_{jk} \sim \mathcal{N}(0, 1/m)$ sketches each whitened direction to $m = 2048$ dimensions, then a thin SVD of the mean-centered sketches in the candidate pool yields the top-k right singular vectors $\boldsymbol{V}_k \in \mathbb{R}^{m \times k}$:

$$\widetilde{\boldsymbol{z}}_i \;=\; \boldsymbol{G}\,\boldsymbol{u}_i, \qquad \boldsymbol{z}_i \;=\; \left(\widetilde{\boldsymbol{z}}_i - \bar{\widetilde{\boldsymbol{z}}}\right) \boldsymbol{V}_k \;\in\; \mathbb{R}^k. \tag{31}$$

PCA retains the dominant empirical covariance directions of the centered Fisher-whitened sketches. We use $m = 2048$ for the JL sketch and retain $k = 8$ PCA directions. In implementation, the pairwise term C_{ij} is evaluated as $\boldsymbol{z}_i^\top \boldsymbol{z}_j$.

Information, alignment, sample scores, and the top-K shortlist are fixed before dimensionality reduction; the reduced features are used to construct the batch redundancy kernel. Projection diagnostics and component ablations are reported in Appendices F.3 and F.4.

Hardware and runtime. All training runs are executed on a single NVIDIA B200 GPU. For the runtime comparison, we use the verified wall-clock times reported in Section 5.3; the same timing protocol is applied to all methods.

F.2 Evaluation Protocol

We evaluate each fine-tuned model on three representative capabilities: general knowledge, code generation, and mathematical reasoning. The main results in Table 1 report MMLU, HumanEval, and GSM8K, and the *Average* row is the unweighted mean of the three benchmark scores.

Benchmarks. For general knowledge and multitask understanding, we use MMLU [16], which contains $14{,}042$ multiple-choice questions across 57 subjects. We evaluate MMLU in the zero-shot setting and report multiple-choice accuracy.

For mathematical reasoning, we use GSM8K [11], which contains $1{,}319$ grade-school math word problems. We evaluate GSM8K with 4-shot setting and report exact match on the final numerical answer.

For code generation, we use HumanEval [8], which contains 164 Python programming problems. We evaluate HumanEval with one greedy completion per problem and report PASS@1 based on execution against the provided unit tests.

We use standard evaluation implementations: MMLU and GSM8K are evaluated with LM-EVAL-HARNESS [14], while HumanEval is evaluated following the official implementation.

Metrics. For MMLU, accuracy is computed as the fraction of questions for which the model selects the correct answer option:

$$\mathrm{ACC} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\{\hat{a}_i = a_i\}.$$

For HumanEval, PASS@1 is the fraction of problems for which the single generated completion passes all unit tests:

$$\text{PASS@1} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\{\text{completion } i \text{ passes all tests}\}.$$

For GSM8K, exact match is computed after extracting the final numerical answer:

$$\mathrm{EM} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\{\hat{y}_i = y_i\}.$$

Decoding and execution. For MMLU, no free-form generation is used; answer choices are scored directly and the option with the highest model likelihood is selected. For HumanEval, we use greedy decoding without sampling, prompt the model in a code-continuation format, and truncate the generation at standard Python stop sequences. Each generated program is executed in an isolated subprocess with a timeout, and a problem is counted as correct only if all provided tests pass. For GSM8K, we use greedy decoding and extract the final numerical answer from the generated solution.

F.3 Projection Diagnostics

We report the mean absolute error (MAE) in pairwise cosine similarities relative to the unprojected Fisher-whitened features.

Table 3: Projection diagnostics.

Diagnostic	Result
JL-2048: cosine MAE	0.0178
PCA-8: redundancy discount (Llama)	0.04
PCA-8: redundancy discount (Qwen)	0.24

F.4 Core Component Ablations

Table 4 compares individual components on Llama-3.1-8B while keeping the other settings fixed. Removing the gate or Fisher-cosine normalization lowers the average score, complementing the whitening, PCA, and fusion ablations. The additive comparison is pool-wise mean-matched.

Table 4: Component ablations on Llama-3.1-8B.

Variant	MMLU	HumanEval	GSM8K	Average
Submitted configuration	63.27	40.24	49.05	**50.85**
No Fisher whitening ($F = I$)	62.80	37.20	44.20	48.06
No PCA	63.61	37.80	46.17	49.20
No gate	62.92	40.85	47.08	50.28
Hard gate	62.85	36.59	45.49	48.31
Raw alignment	63.52	35.98	46.63	48.71
Additive, mean-matched fusion	63.17	39.02	45.64	49.28

Soft gating. The sigmoid gate remains positive for negative alignment: such candidates are down-weighted, and can still be selected when their information or complementary direction compensates for that penalty. This explains why GFV can retain some negatively aligned examples while trading off information, alignment, and redundancy.

F.5 Selection and Projection Dimensions

Table 5 varies the shortlist ratio and projection dimensions one at a time on Llama-3.1-8B, keeping all other settings fixed. The submitted configuration uses $K/B = 4$, JL dimension $m = 2048$, and PCA dimension $k = 8$; the no-PCA comparison appears in Table 4.

Table 5: Shortlist and projection-dimension ablations on Llama-3.1-8B.

Setting	MMLU	HumanEval	GSM8K	Average
Submitted configuration	63.27	40.24	49.05	**50.85**
Shortlist ratio $K/B = 1$	63.48	40.24	46.55	50.09
Shortlist ratio $K/B = 2$	63.36	35.37	46.55	48.43
Shortlist ratio $K/B = 8$	63.35	37.80	44.88	48.68
JL dimension $m = 512$	62.97	40.24	45.41	49.54
JL dimension $m = 8192$	63.74	39.02	45.11	49.29
PCA dimension $k = 4$	63.44	37.20	46.55	49.06
PCA dimension $k = 16$	63.54	40.85	47.69	50.69

F.6 Additional Analysis of GFV Selection Dynamics

This appendix provides a post-hoc analysis of the batches selected by GFV during LLAMA3-8B training. At each selection step, the selector evaluates a 120-example candidate pool, forms a 48-example shortlist using the gated sample score, and then chooses a 12-example training batch using the GFV objective. Source labels are used only for analysis and are never provided to the selector.

Notation. For step t, let P_t denote the 120-example candidate pool and let s_i be the scalar score of candidate $i \in P_t$ before applying the GFV set objective. We define the score-only top-k set as

$$T_t^{(k)} = \mathrm{TopK}_{i \in P_t}(s_i, k), \tag{32}$$

and denote the final GFV-selected batch by S_t, where $|S_t| = 12$. The score-only set $T_t^{(12)}$ provides the reference for isolating the effect of GFV's batch-level objective.

Source Composition of GFV Selections Table 6 reports the source composition of GFV-selected examples. The candidate pool closely follows the global corpus mixture, indicating that the sampling

Table 6: Source composition of GFV selections.

Split	Alpaca	CodeAlpaca	ShareGPT	MATH	GSM8K
Global corpus (%)	32.9	25.3	22.8	9.5	9.5
Candidate pool (%)	32.8	25.4	22.8	9.5	9.4
GFV selections (count)	1,491	770	3,268	1,414	965
GFV selections (%)	18.9	9.7	41.3	17.9	12.2

Table 7: Temporal source composition of GFV selections.

Phase	Alpaca	CodeAlpaca	ShareGPT	MATH	GSM8K
Early	19.0%	10.0%	38.6%	17.9%	14.5%
Middle	19.2%	9.9%	41.0%	18.1%	11.8%
Late	18.4%	9.2%	44.5%	17.7%	10.3%

stream itself is not source-skewed. In contrast, the final GFV selections substantially differ from the corpus prior: ShareGPT and MATH are enriched, while Alpaca and CodeAlpaca are down-weighted. This shows that GFV does not simply preserve the original data mixture, but induces a selected curriculum shaped by sample scoring and set-level geometry.

Temporal Source Curriculum We next examine how the selected source mixture evolves over training. Table 7 divides the GFV selections into three temporal phases. ShareGPT remains the largest selected source throughout training and increases from 38.6% in the early phase to 44.5% in the late phase. MATH is also consistently over-represented relative to its corpus prior, while CodeAlpaca remains strongly under-selected across all phases. These trends suggest that GFV maintains a stable preference for conversational and reasoning-heavy examples, rather than simply tracking the most frequent sources in the corpus.

Effect of the Gate Activation Boundary The alignment gate is disabled during the initial stage of training and activated afterwards. We therefore report selections before and after this boundary separately. As shown in Table 8, the active phase continues to favor ShareGPT and MATH. Compared with the active-phase candidate pool, ShareGPT increases from 22.9% to 41.5%, and MATH increases from 9.5% to 17.5%. This indicates that the source shift is not confined to the initial stage, but persists after the gate becomes active.

GFV Re-ranks Score-Only Selection We next quantify how much GFV changes the selected batch relative to score-only top-12 selection. For each step, define

$$\Delta_t = |S_t \setminus T_t^{(12)}| = 12 - |S_t \cap T_t^{(12)}|, \tag{33}$$

where Δ_t counts the number of GFV-selected examples that are not contained in the score-only top-12 set from the same candidate pool. This comparison isolates the contribution of the batch-level GFV objective: score-only selection ranks examples independently by s_i, whereas GFV additionally accounts for redundancy among Fisher-whitened update directions.

Table 9 shows that GFV substantially re-ranks the score-only selection. After the gate is activated, GFV replaces 3.02 examples per step on average, corresponding to 25.2% of the selected batch. Moreover, 98.8% of active-phase steps contain at least one replacement. Thus, GFV is not a minor perturbation of scalar-score ranking; it consistently changes the selected set through its batch-level objective.

The replacement rate is highest immediately after gate activation and then gradually stabilizes. Across 100-step windows after activation, the mean number of replacements is 4.03 for steps 100–199, 3.19 for steps 200–299, 2.97 for steps 300–399, 2.75 for steps 400–499, 2.46 for steps 500–599, and 2.54 for steps 600–658. This pattern suggests that GFV induces the largest set-level changes immediately after gate activation, and then continues to make smaller but persistent adjustments throughout training.

Table 8: Source composition before and after gate activation.

Phase	Alpaca	CodeAlpaca	ShareGPT	MATH	GSM8K
Gate off	15.1%	7.5%	40.3%	20.3%	16.8%
Gate on	19.5%	10.1%	41.5%	17.5%	11.4%

Table 9: Score-only top-12 versus GFV selection. A replacement is a GFV-selected example outside the score-only top-12 set.

Phase	Steps	Overlap	Repl.	Med.	Any repl.
Gate off	100	10.28	1.72	2	96.0%
Gate on	559	8.98	3.02	3	98.8%
All	659	9.18	2.83	3	98.3%

What GFV Adds and Removes Table 10 reports the source-level effect of GFV relative to score-only top-12 selection in the active phase. GFV shifts probability mass away from Alpaca and CodeAlpaca and toward ShareGPT and MATH. Among examples added by GFV, 49.3% come from ShareGPT and 21.2% from MATH. Conversely, among examples removed from the score-only top-12 set, 36.3% come from Alpaca and 17.3% from CodeAlpaca. This source-level shift is consistent with the overall composition analysis above.

Table 11 further compares the scalar and component scores of these sets. GFV-selected batches have a slightly lower mean scalar score than score-only top-12 batches, which is expected because GFV optimizes a set objective rather than selecting examples independently by scalar score. However, the final GFV batches have higher mean information than score-only top-12 batches. The examples added by GFV also have substantially higher information than the examples removed from the score-only set, despite having lower mean scalar scores. This pattern supports the interpretation that GFV trades off individual scalar score against batch-level volume and complementarity.

G Further Discussion

This section situates our framework in the broader landscape of adaptive optimization, data attribution, and information geometry.

G.1 Shared Geometry, Different Roles: Relation to Adaptive Optimizers

The information term $I_i = \boldsymbol{g}_i^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i$ is algebraically identical to the squared preconditioned step size of an Adam-like optimizer [25]. This resemblance is expected and instructive: both quantities arise from the same diagonal Fisher preconditioner, but serve fundamentally different purposes.

Adam maintains running estimates $\boldsymbol{m}_t = \beta_1 \boldsymbol{m}_{t-1} + (1-\beta_1)\boldsymbol{g}_t$ and $\boldsymbol{v}_t = \beta_2 \boldsymbol{v}_{t-1} + (1-\beta_2)\boldsymbol{g}_t^2$, and updates parameters by $\Delta_t^{\mathrm{Adam}} = -\eta\, \boldsymbol{m}_t / (\sqrt{\boldsymbol{v}_t} + \epsilon)$. Dropping momentum and identifying $\boldsymbol{v}_t \approx \mathrm{diag}(\widetilde{\boldsymbol{F}}_t)$ gives

$$\tfrac{1}{\eta^2} \|\Delta_t^{\mathrm{Adam}}\|_2^2 \;\approx\; \boldsymbol{g}_t^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_t \;=\; I_t.$$

The leading-order match reflects the shared preconditioner, but three structural gaps remain:

1. **Data valuation vs. update execution.** I_i is a directionless scalar that ranks candidate samples by informational value to decide *which* data to train on. The Adam update is a d-dimensional vector that determines *how* parameters move once the batch has been fixed. In other words, adaptive optimization decides how to move once the force is given; our criterion decides which forces should enter the update at all.
2. **Per-sample score vs. aggregated state.** I_i pairs the shared preconditioner with the gradient of a *single* candidate, producing a score that can be compared across all candidates at step t. Adam's update blends the entire gradient history in both numerator ($\boldsymbol{m}_t$) and denominator ($\sqrt{\boldsymbol{v}_t}$) and cannot be decomposed into per-sample contributions.

Table 10: Active-phase source composition of score-only top-12 selection, final GFV selection, and replaced examples.

Set	Total	Alpaca	CodeAlpaca	ShareGPT	MATH	GSM8K
Score top-12	6,708	25.5%	12.3%	35.3%	14.0%	12.9%
GFV final	6,708	19.5%	10.1%	41.5%	17.5%	11.4%
Added by GFV	1,690	12.5%	8.8%	49.3%	21.2%	8.1%
Removed from score top-12	1,690	36.3%	17.3%	24.8%	7.5%	14.1%

Table 11: Active-phase metric averages for score-only and GFV-selected sets.

Set	Mean score	Mean information	Mean alignment
Score top-12	8.226	7.41M	1.6k
GFV final	8.142	8.43M	1.1k
Added by GFV	7.706	6.53M	-0.5k
Removed from score top-12	8.040	2.47M	1.8k

3. **Logarithmic saturation.** Under the Laplace–Fisher–Gauss–Newton chain (Appendix B), the exact information gain is $\mathrm{IG}_i = \frac{1}{2}\log(1 + I_i)$, which is strictly concave in I_i: very large parameter perturbations yield diminishing informational returns. Adam's recursion contains no analogous saturation mechanism.

The fundamental difference is one of purpose: adaptive optimization decides how to execute a parameter update; our criterion decides which candidate updates are worth executing at all. The log-determinant batch objective reinforces this distinction—it does not change the optimizer's update rule but selects a set of candidate forces whose Fisher-whitened directions span a large, non-redundant volume.

G.2 Prospective Valuation vs. Retrospective Attribution: Relation to Influence Functions

Influence functions [26] ask how upweighting a training example changes the loss at a separate query point. In its classical form,

$$\mathcal{I}(z_i, z_q) = -\nabla_{\boldsymbol{\theta}}\ell(z_q; \hat{\boldsymbol{\theta}})^\top \boldsymbol{H}_{\hat{\boldsymbol{\theta}}}^{-1} \nabla_{\boldsymbol{\theta}}\ell(z_i; \hat{\boldsymbol{\theta}}),$$

where $\boldsymbol{H}_{\hat{\boldsymbol{\theta}}}$ is the Hessian at a stationary empirical-risk minimizer. Our alignment statistic $A_i = \bar{\boldsymbol{g}}_t^\top \widetilde{\boldsymbol{F}}_t^{-1} \boldsymbol{g}_i$ has a similar bilinear shape, but the resemblance is limited.

Influence functions are *retrospective*: they ask how a training point would affect a query after the model has been fitted. Our criterion is *prospective*: it asks whether selecting a candidate now would produce an informative and descent-aligned update at the current nonstationary iterate.

The two quantities also play different roles. Influence is already a signed cross-effect. By contrast, our score separates magnitude from direction: I_i measures how much the sample can change the local posterior state, while A_i decides whether that change is compatible with optimization progress. This separation matters because a sample may be highly informative but harmful, or well aligned but uninformative. The batch objective adds a further distinction: influence functions are pairwise sensitivity measures, whereas J_{GFV} is a finite-information volume that discounts repeated Fisher directions across the selected set.

There is also a scope difference. The classical influence derivation is centered at a converged minimizer where the first-order condition $\nabla\hat{L}(\hat{\boldsymbol{\theta}}) = 0$ holds; at a nonstationary iterate the formula loses its formal justification. Our information gain is defined as a prospective design value at every $\boldsymbol{\theta}_t$ along the training trajectory—a critical distinction for online selection, which operates precisely in the pre-convergence regime.

G.3 Variational and Higher-Order Geometric Perspectives

Our framework admits a deeper geometric reading that connects information geometry to geometric mechanics. [28] showed that divergence functions on a statistical manifold and discrete Lagrangians

in geometric mechanics share the same algebraic role: both are real-valued functions on $\mathcal{M} \times \mathcal{M}$ that generate connections, metrics, and symplectic forms through cross-differentiation. Concretely, a statistical divergence $D(p\|q)$ matches the exact discrete Lagrangian to third order if and only if the manifold carries a *Hessian structure* [41, 34]—i.e., the metric is locally the Hessian of a convex potential. Exponential families satisfy this condition with the log-partition function as potential and the Fisher metric as the induced Hessian.

Since $\mathrm{IG}_i = D_{\mathrm{KL}}(p_{t+1}^{(i)} \| p_t)$ is itself a divergence between neighboring points on the model manifold, the Leok–Zhang result implies that our information-gain proxy is a third-order-accurate discrete Lagrangian for the natural dynamics of the parameter space. Under this reading, $\log(1 + I_i)$ is the local action of the candidate motion, $\phi(A_i)$ enforces admissibility to the descent half-space, and the batch objective J_{GFV} is the discrete action over the selected set—additive when Fisher directions are independent, saturating when they overlap.

Fisher information defines only the second-order metric of model distinguishability. The same metric supports distinct affine connection structures, encoded by the Amari–Chentsov cubic tensor [4, 10]

$$C_{ijk}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\theta}}[\partial_i \ell \, \partial_j \ell \, \partial_k \ell] \,,$$

which generates α-connections via $\Gamma_{ijk}^{(\alpha)} = \Gamma_{ijk}^{(0)} - \frac{\alpha}{2} C_{ijk}$. Our scores I_i and A_i are purely second-order; incorporating the cubic tensor could, in principle, distinguish updates along exponential geodesics from those along mixture geodesics. We discuss the resulting extensions—dual formulations via e/m-duality, discrete Noether symmetries, and third-order selection rules—as future work in Appendix I.3.

H Extended Related Work

H.1 Online and Dynamic Data Selection

Offline selection methods choose training subsets before optimization begins, using distribution matching, curation, or gradient similarity [48, 51, 46, 30]; see [1] for a survey. These approaches are complementary to ours but do not adapt to the evolving model state.

Online methods re-evaluate data at every step. Loss- and gradient-based approaches prioritize examples by response magnitude: Online Batch Selection [31] and Selective Backprop [22] rank by loss; importance-sampling methods reweight by gradient norm [23]. A second line of work uses training dynamics to separate genuinely hard examples from noisy or irreducible ones. Dataset Cartography [43] tracks confidence variability, RHO-Loss [38] selects on reducible loss, Recency Bias [42] uses short-horizon uncertainty, and LEAD [29] fuses instantaneous and smoothed loss signals. GREATS [45] and UDS [52] further introduce batch-level Taylor utilities and utility–diversity trade-offs. Despite these advances, most methods still equate response strength or representation diversity with training value.

Information- and geometry-based selection is closest to our work. FisherSFT [12] scores SFT examples by Fisher-based information gain; SPICE [6] uses log-determinant Fisher information with conflict-aware penalties; GIST [37] employs gradient subspaces and projected alignment for targeted instruction tuning. We share the view that sample value should be assessed through the induced update, but differ in three respects: (i) the sample score is derived from local parameter-space information gain at the current iterate; (ii) it is gated by descent alignment, since informative updates can still be counterproductive; and (iii) it is lifted to a calibrated batch-level Fisher-volume objective that preserves singleton scores while discounting redundant Fisher directions.

H.2 Information Geometry and Bayesian Data Acquisition

The Fisher–Rao metric equips a parametric family with a reparameterization-invariant notion of distance derived from the KL divergence [39, 10, 4]. Its main use in optimization is natural gradient descent, which preconditions the gradient by the inverse Fisher matrix [3, 35], with practical approximations such as K-FAC [36]. The same geometry underlies trust-region policy optimization in reinforcement learning [40] and natural-gradient variational inference [19, 24]. These works use Fisher geometry to decide *how* to move parameters; we use it one step earlier, to decide *which data-induced movements are worth taking*.

Bayesian experimental design provides the information-theoretic foundation for data acquisition: selecting observations that maximize expected posterior utility, typically measured by entropy reduction or KL divergence [33, 5]. Related log-determinant and submodular objectives appear in sensor placement and Bayesian optimization [27, 18, 2]. Our setting differs in that the "observations" are training samples consumed during optimization, so the acquisition criterion must additionally enforce descent alignment and batch-level non-redundancy.

I Limitations and Broader Impact

I.1 Limitations

The proposed framework still relies on local approximations to posterior change and descent behavior, and its practical quality depends on surrogate quantities such as the damped Fisher metric and the reference gradient. In large-scale training, these approximations can be imperfect, and their fidelity may vary across architectures, optimization regimes, and data mixtures. We therefore view the method as a principled local selection rule rather than a complete description of long-horizon training dynamics.

I.2 Broader Impact

A better online data-selection rule can reduce training cost and improve sample efficiency, which may make large-model fine-tuning more accessible. At the same time, any selection mechanism that systematically prioritizes some training signals over others can also amplify existing dataset imbalances or underrepresent rare but important behaviors. In practice, data-selection systems should therefore be used together with dataset auditing, held-out evaluation across diverse capabilities, and human oversight in high-stakes settings.

I.3 Future Directions

The variational and higher-order geometric connections discussed in Section G.3 suggest several avenues for future research.

Dual formulations via e/m-duality. The e/m-duality of statistical manifolds [4] is the information-geometric counterpart of the Legendre-transform duality between Lagrangian and Hamiltonian mechanics [28]. In discrete mechanics, this duality relates position-based and momentum-based variational integrators. In the data-selection setting, an analogous "Hamiltonian" formulation would express selection criteria in terms of natural parameters and sufficient statistics rather than gradients and Fisher distances, potentially offering computational advantages when the model has a known conjugate structure.

Discrete Noether theorems and conservation laws. Symmetries of a discrete Lagrangian imply discrete conservation laws via Noether's theorem [34]. Since D_{KL} is invariant under sufficient-statistic reparametrizations, the Leok–Zhang correspondence raises the question of whether information-geometric selection rules preserve analogous invariants along training trajectories—for instance, whether batches selected by J_{GFV} conserve certain information-geometric momenta or volume forms.

Third-order selection rules. Incorporating the Amari–Chentsov cubic tensor C_{ijk} into the selection criterion could yield rules that distinguish between updates along exponential geodesics (natural gradient paths) and mixture geodesics (moment-matching paths) [4, 10]. Such third-order extensions would be particularly relevant for models with strong non-Gaussian posterior geometry, where the local quadratic approximation underlying our current framework is least accurate.

Beyond the Hessian manifold assumption. The third-order matching between divergences and discrete Lagrangians holds exactly on Hessian manifolds (e.g., exponential families) [41]. For general deep networks, the parameter space is not globally Hessian, and quantifying how the variational fidelity degrades—and whether it can be restored by working in a suitable function-space geometry—remains an open problem.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction state the paper's contributions as update-centric online data selection in parameter space, Fisher-whitened information scoring, descent alignment gating, and the Gated Fisher Volume batch objective. These claims are supported by the theoretical results in Sections 3–4 and Appendices B–E, and by the fine-tuning experiments in Section 5.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

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

   Answer: [Yes]

   Justification: Appendix I.1 discusses limitations arising from local approximations to posterior change and descent behavior, the use of a damped Fisher surrogate, and dependence on the quality of the reference gradient. It also states that the method should be viewed as a principled local selection rule rather than a complete description of long-horizon training dynamics.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

   Justification: The assumptions for the theoretical results are stated with the corresponding results, including the local Gaussian posterior approximation, Gauss–Newton/Fisher approximation, smoothness assumptions, nonnegative calibration, and the local linear-Gaussian model used for the batch information-gain interpretation. Complete proofs are provided in Appendices B–E.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We provide an anonymized implementation with instructions for environment setup, data preparation, and commands for reproducing the main experimental results. The required public datasets, model backbones, training configuration, and evaluation protocol are described in Section 5 and Appendix F.1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: We provide the code package in the supplementary material, including scripts and instructions for reproducing the main experimental results. The supplementary material also describes the data sources, preprocessing procedure, all hyperparameters, and commands used to run the proposed method.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Section 5 and Appendix F.1 specify the training data, source mixture, model backbones, LoRA setup, optimizer, selection budget, candidate-pool size, selected batch size, shortlist size, Fisher warm-up, gate schedule, sequence lengths, baseline standardization, and evaluation protocol. These details are sufficient to understand the reported experimental comparisons.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

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

   Answer: [No]

   Justification: We do not report error bars or formal statistical significance tests for all main experiments due to the computational cost of repeated LLM fine-tuning runs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Section 5 and Appendix F.1 provide the compute setup used in our experiments, including the GPU type, training configuration, selection budget, and wall-clock runtime protocol.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The research uses public datasets, open model backbones, and automatic benchmark evaluation for algorithmic data-selection research.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: As we discuss in I.2, the method may reduce fine-tuning cost and improve sample efficiency, while also noting that data selection can amplify dataset imbalances or underrepresent rare behaviors. We recommend dataset auditing, diverse held-out evaluation, and human oversight in high-stakes settings.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The paper introduces an algorithmic data-selection method and does not release a new pretrained language model, image generator, or scraped dataset with high misuse risk. The experiments use existing public datasets and open model backbones for evaluation.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We credit the creators and original owners of the existing assets used in this work through citations in Section 5 and use them according to their stated licenses and terms. The training and evaluation assets include ALPACA, CODEALPACA, the ShareGPT-derived dataset `anon8231489123/ShareGPT_Vicuna_unfiltered`, MATH, GSM8K, MMLU, HumanEval, and `lm-eval-harness`. The model and method assets include LLAMA3.1-8B, QWEN3-8B, LoRA, LEAD, and SPICE. The corresponding licenses and terms include Apache License 2.0 for ALPACA, CODEALPACA, the ShareGPT-derived dataset, and QWEN3-8B; MIT License for MATH, GSM8K, LoRA, LEAD, SPICE, MMLU, HumanEval, and `lm-eval-harness`; and the Llama 3.1 Community License Agreement for LLAMA3.1-8B. We respect the license and terms of use associated with each asset.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [Yes]

    Justification: We provide an anonymized supplementary zip file containing the code and configuration files used in the experiments. The package includes documentation for environment setup, data preparation, training, and evaluation commands. Also, a license file is included.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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

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

    Answer: [N/A]

    Justification: The paper does not conduct crowdsourcing experiments or human-subject studies. All experiments use existing datasets, open model backbones, and automatic evaluation benchmarks.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or human-subject research. IRB approval or equivalent human-subject review is therefore not applicable to the reported work.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: LLM assistance was limited to writing, editing, or formatting and did not affect the method, experiments, scientific rigor, or originality of the work.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.