
SAFE-DRIFT: Data Selection for Supervised Fine-tuning with Controllable Off-Target Drifts

Yeo Jin Jung*

Department of Statistics
University of Chicago
yeojinjung@uchicago.edu

Yating Liu*

Department of Statistics
University of Chicago
yatingliu@uchicago.edu

Lalchand Pandia

Department of Computer Science
University of Chicago
lcpandia@uchicago.edu

Claire Donnat

Department of Statistics
University of Chicago
cdonnat@uchicago.edu

Abstract

We propose SAFE-DRIFT, a data-selection framework for supervised fine-tuning that explicitly trades off target improvement against unwanted off target drifts in model behavior. We formalize this objective as a constrained optimization problem that maximizes alignment with the target gradient, subject to a budget on off-target drift which we quantify using the Fisher information of a reference distribution. Under this formalism, the optimal, closed-form update is a damped natural-gradient step that downweights reference-sensitive directions. To estimate these high-dimensional gradients and work within the right geometry, we further derive a scalable algorithm that approximates the Fisher information and the target gradient in a shared low-dimensional subspace. We evaluate SAFE-DRIFT against state-of-the-art data selection methods across diverse domains including coding and medical question answering, yielding competitive target performance while reducing degradation on specified off-target behaviors.

1 Introduction

Supervised fine-tuning (SFT) is an effective technique to adapt large language models (LLMs) to specialized tasks. In many of such tasks, a critical bottleneck is the scarcity of high-quality examples that precisely define the desired behavior. For instance, in medicine, clinician-validated cases may serve as gold-standard target examples; larger biomedical corpora may provide a much wider but noisier training pool [1, 2]. Similarly, in law, a small number of expert-annotated examples may define the target task, while larger collections of legal documents may offer a broader but less aligned training pool [3–5]. In all of these examples, estimates of the target distribution are high-value and expensive to label, whereas the available training pool is broader, cheaper, and only partially aligned with the task of interest. This pool of candidate examples (which we will refer to as the candidate pool) can provide a useful training signal, but training on all of it is not necessarily safe or effective. Carefully selected subsets have indeed been shown to outperform training on the full candidate pool in various settings [6–9] (see Table 2 for additional examples).

At the same time, fine-tuning does not solely affect the capability we intend to improve [10]. Growing evidence shows that seemingly benign finetuning can weaken safety alignment and degrade behavior outside the target task [11, 12]. Beyond safety, fine-tuning can also lead to the forgetting of more general capabilities[13]. We refer to these unintended changes as *off-target drifts*.

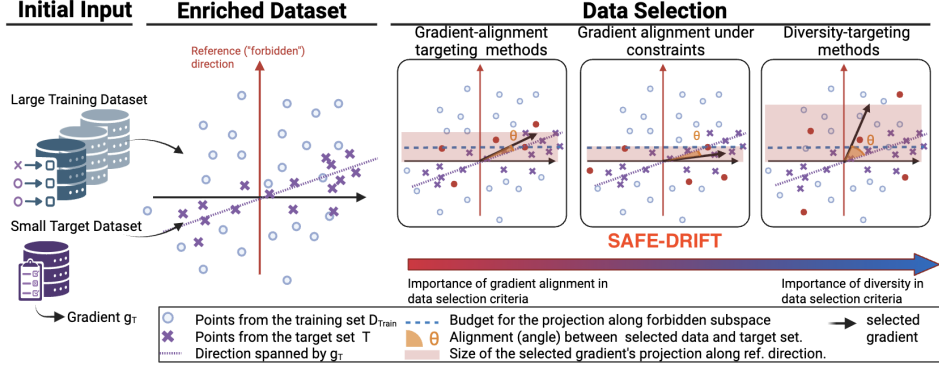


Figure 1: *Trade-offs in data selection: gradient alignment selects candidate gradients aligned with the task but can cause substantial off-target drift (e.g., movement along a "forbidden dimension", in red). Diversity targeting can mitigate this but does not reliably prevent off-target drift. SAFE-DRIFT combines both approaches by constraining updates along the forbidden dimension.*

These observations suggest that fine-tuning is not just a matter of collecting more data, but a selecting the right ones: given a large and noisy candidate pool, we must choose examples that improve the target task without inducing unnecessary off-target drift. The key question is therefore: *which examples from the candidate pool should we train on in order to improve the target task while safeguarding other model behaviors?*

Existing data selection methods broadly fall into two (and often mutually exclusive) families. The first selects candidate examples that appear most valuable for the target task, either through explicit quality proxies [14–17] or through direct influence estimation [6, 18]. The second prioritizes diversity or distributional coverage of selected candidates to improve generalization [19, 20]. We refer the reader to [21] for a comprehensive review. None of these methods explicitly constrains the off-target drift induced by the selected subset, nor do they account for the possible entanglement between target-improving and reference-sensitive parameter directions [22–25].

If the gradients that yield the largest improvement in the target task overlap with directions associated with worse performance on a reference task, then aggressively selecting them can induce substantial off-target drift. Diversity-based selection, on the other hand, can dilute these update antagonisms but may not deliver the best target performance. Optimal data selection must therefore navigate potential tensions between target improvement and reference degradation. Here, we propose an algorithm that explicitly models data selection as a constrained optimization problem to limit unwanted movement along reference-sensitive directions, as illustrated in Figure 1.

Contributions. We summarize our contributions as follows:

- We introduce **SAFE-DRIFT, a new framework to select data for fine-tuning under an explicit drift budget**. We represent drifts via the reference Fisher information, yielding a principled geometric framework where certain parameter directions incur higher costs because they more strongly alter a reference model behavior (Section 2).
- **A connection between target-aligned and diversity-based selection**. We show that the optimal update under a reference-drift budget is a damped natural-gradient direction. This clarifies when target-aligned selection cannot safeguard against off-target drift, when diversity can reduce the drift, and how both arise as limiting or special cases of a single constrained update problem (Section 2.1).
- **A scalable low-rank implementation**. We build a shared low-rank basis that merges leading reference-sensitive directions with residual task-relevant directions (Section 2.3), making selection tractable and enabling diagnostics of target–reference entanglement.
- **Experiments demonstrating controllable tradeoffs**. Across coding, medical question-answering, emotional support dialogues, and instruction-following experiments, SAFE-DRIFT achieves competitive target gains while reducing off-target drift on the specified reference behaviors. Ablations validate the roles of the drift budget and low-rank approximation (Section 3 and Appendix D.4).

2 Method

Consider a base language model with parameters $\theta_0 \in \mathbb{R}^d$ that defines a conditional distribution over output tokens $y \in \mathcal{Y}$ given input tokens $x \in \mathcal{X}$. Let $z = (x, y)$ denote an input–output pair, and let $\ell(z; \theta)$ denote the training loss (typically the negative log-likelihood of the output sequence, or of a single answer token in a multiple-choice setting). We assume that we are given a large candidate training pool $\mathcal{D}_{\text{train}} = \{z_i\}_{i=1}^{N_C}$ and a small target set $\mathcal{D}_{\text{tar}} = \{z_j\}_{j=1}^{N_T}$ with $N_T \ll N_C$ that encodes the capability we wish to improve. We are also given a reference set $\mathcal{D}_{\text{ref}} = \{z_i\}_{i=1}^{N_R}$ of prompts from a reference distribution $\mathcal{R}$ over inputs, on which the model’s output distribution should be preserved. Let the target risk at parameter $\theta \in \mathbb{R}^d$ be defined as:

$$L_{\mathcal{T}}(\theta) := \mathbb{E}_{z \sim \mathcal{T}} [\ell(z; \theta)] = \mathbb{E}_{x \sim \mathcal{T}_X} \mathbb{E}_{y \sim q_{\mathcal{T}}(\cdot | x)} [\ell((x, y); \theta)], \quad (1)$$

where $\mathcal{T}_X$ is the marginal of the target distribution $\mathcal{T}$ on inputs and $q_{\mathcal{T}}(\cdot | x)$ denotes the gold-standard conditional distribution. Our goal is to select a subset $\mathcal{S} \subset \mathcal{D}_{\text{train}}$ of size $|\mathcal{S}| = S$ such that fine-tuning on $\mathcal{S}$ improves the target $L_{\mathcal{T}}(\theta)$ while preserving the model’s output distribution on $\mathcal{R}$.

We proceed in two stages. We first characterize the optimal gradient step, which minimizes $L_{\mathcal{T}}(\theta)$ while preserving behavior under $\mathcal{R}$. We then identify a subset $\mathcal{S} \subset \mathcal{D}_{\text{train}}$ of size S whose aggregate effect most closely matches the optimal update. The model is finetuned on $\mathcal{S}$ in one pass, without any online reweighting or iterative refinement. The first subsection characterizes the oracle update, assuming access to the full target and reference populations. We then derive the subset-selection algorithm and its empirical implementation using the finite datasets $\mathcal{D}_{\text{train}}$, $\mathcal{D}_{\text{tar}}$, and $\mathcal{D}_{\text{ref}}$.

2.1 Optimal parameter update under a drift budget

We begin with a population-level oracle problem in which both the population gradient of the target loss $g_{\mathcal{T}} = \mathbb{E}[\nabla_{\theta} L_{\mathcal{T}}(\theta_0)]$ and the reference distribution $\mathcal{R}$ are known exactly. Here $\theta_0 \in \mathbb{R}^d$ denotes the base model parameters, and the finetuned model is parametrized by $\theta = \theta_0 + \Delta\theta$ for some update $\Delta\theta \in \mathbb{R}^d$. The oracle question we ask is: *what is the optimal parameter update $\Delta\theta$ that maximally improves target performance while safeguarding against off-target drift?*

Off-target drift on reference distribution. For the finetuned model $\theta_0 + \Delta\theta$, define the expected drift on the reference distribution as the Kullback-Leibler divergence with the base model,

$$D_{\mathcal{R}}(\Delta\theta) := \mathbb{E}_{x \sim \mathcal{R}} [D_{\text{KL}}(p_{\theta_0}(\cdot | x) \| p_{\theta_0 + \Delta\theta}(\cdot | x))]. \quad (2)$$

The quantity $D_{\mathcal{R}}(\Delta\theta)$ captures the off-target effects of fine-tuning: how much the parameter update $\Delta\theta$ alters model behavior on $\mathcal{R}$, outside the target task. We note here that $D_{\mathcal{R}}(\Delta\theta)$ is defined globally for any $\Delta\theta$. However, because our results consider small fine-tuning updates, it is natural to study the local behavior of $D_{\mathcal{R}}(\Delta\theta)$ around $\Delta\theta = 0$. We next show that, to second order, off-target drift is a quadratic form induced by the Fisher information of the model on the reference distribution.

Definition 1 (Reference Fisher). Let $s_{\theta_0}^R(x, y) := \nabla_{\theta} \log p_{\theta_0}(y | x)$. The *reference Fisher information* at θ_0 is:

$$\mathbf{F}_{\mathcal{R}} := \mathbb{E}_{x \sim \mathcal{R}} \mathbb{E}_{y \sim p_{\theta_0}(\cdot | x)} [s_{\theta_0}^R(x, y) s_{\theta_0}^R(x, y)^{\top}]. \quad (3)$$

Proposition 1 (Local reference behavior). *Assume standard smoothness and integrability conditions so that differentiation may be exchanged with expectation in a neighborhood of θ_0 . Then, as $\Delta\theta \rightarrow 0$,*

$$D_{\mathcal{R}}(\Delta\theta) = \frac{1}{2} \Delta\theta^{\top} \mathbf{F}_{\mathcal{R}} \Delta\theta + o(\|\Delta\theta\|_2^2). \quad (4)$$

The smoothness conditions and proof are provided in Appendix B.1. Proposition 1 shows that, for small fine-tuning updates, the cost $D_{\mathcal{R}}(\Delta\theta)$ is estimated as a quadratic form with the curvature given by $\mathbf{F}_{\mathcal{R}}$, rather than the classic Euclidean norm $\|\Delta\theta\|_2^2$: directions with large curvature under $\mathbf{F}_{\mathcal{R}}$ induce larger changes in the model’s output distribution on $x \sim \mathcal{R}$, whereas directions lying near the null space of $\mathbf{F}_{\mathcal{R}}$ do not increase the off-target drift.

Target gain as gradient alignment. Recall the population gradient loss of the target at the base model θ_0 : $g_{\mathcal{T}} := \nabla_{\theta} L_{\mathcal{T}}(\theta_0)$. Without loss of generality, throughout the rest of this paper, we will assume $g_{\mathcal{T}}$ to have unit norm $\|g_{\mathcal{T}}\|_2 = 1$. For a small update $\Delta\theta$, a first order Taylor approximation at θ_0 gives,

$$L_{\mathcal{T}}(\theta_0 + \Delta\theta) - L_{\mathcal{T}}(\theta_0) = g_{\mathcal{T}}^{\top} \Delta\theta + o(\|\Delta\theta\|_2). \quad (5)$$

It follows that the update providing the steepest decrease in target loss is proportional to $g_{\mathcal{T}}$. However, this strategy might induce updates in the space spanned by $\mathbf{F}_{\mathcal{R}}$, creating a potential tradeoff between maximizing target accuracy, and minimizing off-target drift.

The oracle update problem. We construct a bridge of data selection by formulating it as a constrained optimization problem of $\Delta\theta$: we maximize the target-side gain $-g_{\mathcal{T}}^{\top}\Delta\theta$, while allowing only a small budget of off-target drift $\frac{1}{2}\Delta\theta^{\top}\mathbf{F}_{\mathcal{R}}\Delta\theta$ on the reference distribution. This leads to the following oracle problem: *among all small parameter updates, choose the one that maximizes target gain while spending at most ρ , a fixed drift budget*:

$$\Delta\theta^* \in \arg \max_{\Delta\theta \in \mathbb{R}^d} \left\{ -g_{\mathcal{T}}^{\top}\Delta\theta \right\} \quad \text{subject to} \quad \frac{1}{2}\Delta\theta^{\top}\mathbf{F}_{\mathcal{R}}\Delta\theta \leq \rho, \quad \|\Delta\theta\|_2 \leq \epsilon. \quad (6)$$

There are two central parameters: ρ , which sets the budget on off-target drift, and ϵ which constrains the magnitude of the update.

The constraint ϵ serves two purposes: it ensures that $\Delta\theta$ is small enough for the local approximations in 4 and 5 to hold, and it regularizes the update in directions where $g_{\mathcal{T}}$ lies in the null space of $\mathbf{F}_{\mathcal{R}}$. In practice, we assume ϵ is given. As we show in Appendix B.4, ϵ is determined by the optimizer’s learning rate η . For stochastic gradient descent, for instance, we have $\|\Delta\theta\|_2 \leq \eta/\sqrt{S}\|\mathbf{G}\|_{\text{op}}$ where S is the selection budget and $\|\mathbf{G}\|_{\text{op}}$ is the spectral norm of the candidate gradient matrix, both quantities that are fixed characteristics of the problem.

Consequently, the sole quantity in 6 that must be specified is ρ , the maximum off-target drift we are willing to tolerate. To make this quantity interpretable, let $B_x \subseteq \mathcal{Y}$ be a set of undesirable outputs for prompt x , for instance, a set of toxic or stereotyped responses. By Pinsker’s inequality,

$$[p_{\theta}(B_x|x) - p_{\theta_0}(B_x|x)]_+ \leq |p_{\theta}(B_x|x) - p_{\theta_0}(B_x|x)| \leq \sqrt{\frac{1}{2}D_{KL}(p_{\theta_0}(\cdot|x), p_{\theta}(\cdot|x))} \leq \sqrt{\frac{\rho}{2}}$$

Therefore, for any undesirable output event $B_x \subseteq \mathcal{Y}$, the expected increase in probability of B_x under the finetuned model relative to the base model (over reference prompts) is bounded by $\sqrt{\rho/2}$. In practice, we treat ρ as a hyperparameter that requires tuning to efficiently safeguard against deleterious off-target drifts; we ablate over a range of ρ ’s in Section 3. We now provide the optimal oracle solution of 6.

Proposition 2 (Oracle Update $\Delta\theta^*$). *Assume the target gradient vector satisfies $g_{\mathcal{T}} \neq 0$. When both the drift budget ρ and the norm budget ϵ are active, the solution to (6) is,*

$$\Delta\theta^* = -\epsilon \frac{(\mathbf{F}_{\mathcal{R}} + \alpha\mathbf{I})^{-1}g_{\mathcal{T}}}{\|(\mathbf{F}_{\mathcal{R}} + \alpha\mathbf{I})^{-1}g_{\mathcal{T}}\|_2},$$

where α is chosen to satisfy $\frac{1}{2}(\Delta\theta^*)^{\top}\mathbf{F}_{\mathcal{R}}\Delta\theta^* = \rho$.

The remaining cases, in which one or both constraints are loose, are characterized in Appendix B.2. In particular, the maximum-gain direction is proportional to the damped natural gradient of the target objective under the reference distribution.

A complete characterization of all cases, along with proofs and an explicit formula for α as a function of (ϵ, ρ) , is provided in Appendices B.2 and B.3.

Remark 1 (Connection to existing data selection methods). Proposition 2 places existing data selection methods within a common framework and helps explain the conditions under which each paradigm is preferred. When $\rho \rightarrow \infty$, the drift constraint is never active and the optimal update reduces to $\Delta\theta^* \propto -g_{\mathcal{T}}$, recovering pure gradient alignment with the target—the principle underlying LESS [6]. This connection also reveals when target-aligned selection can be insufficient: if $g_{\mathcal{T}}$ has large components along the leading eigenvectors of $\mathbf{F}_{\mathcal{R}}$ (“entanglement”), then selecting for gradient alignment with $g_{\mathcal{T}}$ necessarily induces large off-target drift. In this regime, diversity-based selection can be preferable as the parameter update is not necessarily concentrated along $g_{\mathcal{T}}$ and may avoid off-target-sensitive directions even without explicitly modeling them.

2.2 Algorithm: Safe-Drift Data Selection

Let $w \in \{0,1\}^{N_C}$ denote the selection vector, where $w_i = 1$ indicates that candidate example i is selected for fine-tuning. Let $\mathbf{G} \in \mathbb{R}^{d \times N_C}$ be the matrix of candidate gradient updates. The

parameter update induced by most gradient-based data selection methods can be summarized as: $\Delta\theta(w) = -\eta \mathbf{P} \mathbf{G} w / |\sum_i w_i|$, where $\mathbf{P} \in \mathbb{R}^{d \times d}$ denotes the preconditioner induced by the optimizer and $\eta > 0$ is the learning rate. For example, $\mathbf{P} = \mathbf{I}_d$ recovers stochastic gradient descent, while adaptive methods such as Adam [26] use a diagonal approximation for local curvature. We defer a further discussion of the choice of $\mathbf{P}$ to Appendix B.4. In our setting, the goal is to approximate the optimal safe update $\Delta\theta^*$ in Proposition 2 that maximizes the target gain while satisfying the constraint on off-target drift. Accordingly, we aim to construct a subset of exactly S examples by,

$$w^* = \arg \min_{w \in \{0,1\}^{N_C}, \mathbf{1}^\top w = S} \|\Delta\theta(w) - \Delta\theta^*\|_{\mathbf{F}_R + \alpha \mathbf{I}}^2, \quad (7)$$

where $\|x\|_{\mathbf{F}_R + \alpha \mathbf{I}}^2 := x^\top (\mathbf{F}_R + \alpha \mathbf{I}) x$ for a vector $x \in \mathbb{R}^d$.

To solve (7) in practice, we use a greedy forward selector, following the common use of greedy matching procedures in sparse approximation [27] and gradient-based subset selection [28]. We detail the full pipeline in Algorithm 1 below and Algorithm 3 in Appendix C.2.

Remark 2. We carry out the reconstruction within the geometry defined by the reference set, which enables us to assign higher penalties to reconstructions that have larger components along “costly” off-target directions. In Appendix D.4.3, we show that this approach yields stronger safeguarding mechanisms than reconstructions based on the standard Euclidean norm.

First validation on synthetic experiments. To validate our method and gain further intuition, we validate this model on a synthetic linear regression task where target data follow $x \sim \mathcal{N}(0, \Sigma_T)$ and $y = x^\top \beta_T + \xi$ with $\xi \sim \mathcal{N}(0, \sigma^2)$, while reference inputs follow $x \sim \mathcal{N}(0, \Sigma_R)$. We test two regimes: an *entangled setting*, where $-g_T$ has a strong projection on $\mathbf{F}_R$, and a *disentangled setting*, where this projection is weak (details in Appendix D.1). Figure 5 in the Appendix shows that in the entangled regime, SAFE-DRIFT must trade off target gradient alignment against off-target budget due to the coupling between Σ_R and $-g_T$. In the disentangled regime, where $-g_T$ is mostly low-cost, SAFE-DRIFT stays closer to the LESS baseline that maximizes target gain.

2.3 Low-rank estimation of Fisher information and gradients

We turn to the problem of estimating the gradients of the candidates and target, as well as the reference Fisher information required to solve problems (6) and (7). In a very high-dimensional parameter space, plug-in estimation in the full parameter space is computationally unstable, memory-intensive and noisy. In particular, the empirical reference Fisher is rank-deficient whenever the number of reference samples is smaller than the number of trainable parameters. Luckily, prior works have shown that language-model updates often lie in a low-dimensional subspace or admit a low-rank update structure [29–31]. Our goal is therefore to construct a low-rank projection, $\mathbf{U}_K \in \mathbb{R}^{d \times K}$ with $K \ll d$, that captures the reference-sensitive directions while also retaining the dominant task-relevant gradient structure.

Raw estimation. Define the estimated gradient by $\hat{g}_i^{\text{raw}} := \nabla_{\theta} \ell(z_i; \theta_0)$. The target gradient is then estimated by the sample mean $\hat{g}_{\text{tar}}^{\text{raw}} := \frac{1}{N_T} \sum_{z_i \in \mathcal{D}_{\text{tar}}} \nabla_{\theta} \ell(z_i; \theta_0)$. Let $\mathcal{D}_{\text{ref}} = \{x_j\}_{j=1}^{N_R}$ be the held-out set of reference inputs, and let the score vector for the reference data be denoted:

$$s_j^R := \nabla_{\theta} \log p_{\theta_0}(y_j | x_j) \in \mathbb{R}^d, \quad y_j \sim p_{\theta_0}(\cdot | x_j).$$

Let $\mathbf{S}_R \in \mathbb{R}^{N_R \times d}$ denote the matrix of stacked score vectors.

Estimating the low-rank projection $\mathbf{U}_K$. We construct $\mathbf{U}_K$ in two stages.

Step 1: low-rank approximation of $\mathbf{F}_R$. We estimate the leading eigenspace of $\mathbf{F}_R$, and retain the top K_R eigenpairs, $(\hat{\mathbf{U}}_{K_R}, \hat{\Lambda}_{K_R})$, from the raw empirical reference Fisher $\hat{\mathbf{F}}_R^{\text{raw}} = \frac{1}{N_R} \mathbf{S}_R^\top \mathbf{S}_R$.

Step 2: low-rank approximation of target and candidate gradients on the residual subspace.

Since the target and candidate gradients may contain informative structure outside this reference

¹In implementation, we never form the ambient $d \times d$ matrix explicitly. Instead, we work with the smaller $N_R \times N_R$ Gram matrix whose nonzero eigenvalues coincide with those of $\hat{\mathbf{F}}_R^{\text{raw}}$ (see Appendix C.)

eigenspace, we project them onto its orthogonal complement and extract a residual task basis from the projected gradients. Let $\tilde{g}_i$ denote the residualized gradients:

$$\tilde{g}_i := (\mathbf{I}_d - \hat{\mathbf{U}}_{K_R} \hat{\mathbf{U}}_{K_R}^\top) \hat{g}_i^{\text{raw}}, \quad \forall i \in [N_C + N_T]. \quad (8)$$

Stacking the residualized gradients gives $\tilde{\mathbf{G}}_{TC} \in \mathbb{R}^{d \times (N_T + N_C)}$. We then estimate an additional residual basis by retaining the top K_T left singular vectors of $\tilde{\mathbf{G}}_{TC}$, yielding, $\hat{\mathbf{U}}_{K_T} \in \mathbb{R}^{d \times K_T}$.

Stacking the two bases yields the final working basis $\mathbf{U}_K$, which by construction spans both the reference-sensitive and task-relevant directions:

$$\mathbf{U}_K := [\hat{\mathbf{U}}_{K_R} \quad \hat{\mathbf{U}}_{K_T}] \in \mathbb{R}^{d \times K}, \quad K := K_R + K_T, \quad \mathbf{U}_K^\top \mathbf{U}_K = \mathbf{I}_K. \quad (9)$$

We evaluate the effect of the approximation of $\mathbf{F}_R$ and the gradients under this low rank projection in Appendix D.1. As expected, the suitability of the approximation depends on the spectrum of the eigenvalues of $\mathbf{F}_R$ and the singular values of g_T . The selection of K is discussed in Appendix C.1 and D.4.2. Figure 9 and 10 in the Appendix also show that the random projection (such as for instance used in LESS [6]) is less effective to capture spectral structure.

Remark 3 (Entanglement diagnostics). Combining the results from Proposition 2 with the low-rank basis $\mathbf{U}_K$ provides a insight for us to examine target–reference entanglement by decomposing the projected target and candidate gradients into reference-sensitive and residual task components. For example, we record the proportion of target-aligned candidate examples that could induce a drop of performance in the reference task, denoted as $\pi_{Q_{\text{ent}}}$, while we use $\pi_{Q_{\text{clean}}}$ to indicate the proportion of candidates that carry target signal through residual task directions that are less coupled to the reference geometry. We provide their definitions and discuss their interpretation in Appendix D.2.

2.4 Safe update $\hat{\Delta}\theta_K^*$ and selection $\hat{w}$ in practice.

The target and candidate coordinates in the common basis can be calculated by,

$$\hat{b}_i^{(K)} := \mathbf{U}_K^\top \hat{g}_i^{\text{raw}} \text{ for all } z_i \in \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{tar}}, \quad \hat{b}_{\text{tar}}^{(K)} := \frac{1}{N_T} \sum_{i: z_i \in \mathcal{D}_{\text{tar}}} \hat{b}_i^{(K)}. \quad (10)$$

Given α and the scaling factor $\hat{\mu}_K$ to satisfy the reference and norm budgets in (6), which we detail in Appendix C.1, the practical safe update $\Delta\theta^*$ using the low rank basis is,

$$\hat{\Delta}\theta_K^*(\alpha) = -\frac{1}{\hat{\mu}_K} \left(\text{diag}(\hat{\mathbf{\Lambda}}_{K_R}, \mathbf{0}_{K_T}) + \alpha \mathbf{I}_K \right)^{-1} \mathbf{U}_K^\top \hat{g}_{\text{tar}}^{\text{raw}} \in \mathbb{R}^K. \quad (11)$$

Define the reduced preconditioner $\mathbf{P}_K := \mathbf{U}_K^\top \mathbf{P} \mathbf{U}_K \in \mathbb{R}^{K \times K}$, where $\mathbf{P}$ is the preconditioner. Similarly, for the selection vector $w \in \{0, 1\}^{N_C}$ with $\mathbf{1}^\top w = S$, the induced reduced update is,

$$\hat{\Delta}\theta_K(w) = -\frac{\eta}{S} \mathbf{P}_K \mathbf{B}_K w \in \mathbb{R}^K, \quad \mathbf{B}_K := \begin{bmatrix} \hat{b}_1^{(K)} & \dots & \hat{b}_{N_C}^{(K)} \end{bmatrix}. \quad (12)$$

We can now plug in $\hat{\Delta}\theta_K^*(\alpha)$ and $\hat{\Delta}\theta_K(w)$ in (7) to solve for the final selection $\hat{w}$. In practice, we turn the norm objective to a score for each candidate and select the top S points as detailed in Algorithm 1 and Appendix C.1.

Algorithm 1 SAFE-DRIFT data selection in the low-rank space

Input: Damping parameter α , target update $\hat{\Delta}\theta_K^*(\alpha)$, preconditioner $\mathbf{P}_K$, target gradients $\mathbf{B}_K$, learning rate η , subset budget S .

- 1: Form the reduced reference Fisher $\mathbf{M}_{\alpha, K} = \text{diag}(\hat{\mathbf{\Lambda}}_{K_R}, \mathbf{0}_{K_T}) + \alpha \mathbf{I}_K$.
- 2: Whiten the reduced objective via $\mathbf{M}_{\alpha, K}^{1/2}$, and set

$$\tilde{\Delta}_K \leftarrow -\frac{\eta}{S} \mathbf{M}_{\alpha, K}^{1/2} \mathbf{P}_K \mathbf{B}_K, \quad \tilde{\Delta}_K^* \leftarrow \mathbf{M}_{\alpha, K}^{1/2} \hat{\Delta}\theta_K^*(\alpha).$$

- 3: Compute singleton scores for each sample in $\mathcal{D}_{\text{train}}$

$$s_i = \langle \tilde{\Delta}_K(i), \tilde{\Delta}_K^* \rangle - \frac{1}{2} \|\tilde{\Delta}_K(i)\|_2^2, \quad i = 1, \dots, N_C.$$

- 4: Choose the top S candidates using the forward greedy selector (Algorithm 3 in Appendix C.2).

Output: Selection $\hat{w}$, selected subset $\hat{S} := \{i \in [N_C] : \hat{w}_i = 1\}$, and the update $\hat{\Delta}\theta_K(\hat{w})$.

3 Experiments

We compare SAFE-DRIFT against four established data selection baselines. These include (1) random sampling; (2) LESS [6], a target-alignment method which leverages LoRA-based low-dimensional gradient similarity to select data; (3) DSIR [17], a target-alignment method based on hashed n -gram features; (4) Prismatic Synthesis [19], a diversity-based method that clusters candidate gradients and samples from underrepresented clusters.² None of these explicitly controls off-target drift. We additionally compare against two methods that preserve reference behavior: (a) replay, the standard method for catastrophic forgetting [32], and KL-regularized fine-tuning, which penalizes divergence from the base model on reference data.

All experiments use open-weight OLMo-2-7B models (OLMo-2-1124-7B or OLMo-2-1124-7B-Instruct) [33] as the base model with parameter θ_0 ; the robustness study of Section 3.1 additionally evaluates two model families across five scales. We compare the trade-off between target gain and off-target drift, including methods built to preserve reference behavior, and show robustness across model architectures, the drift cost ρ , and the choice of gradient update. The full experimental protocol is summarized in Table 1. We have additional results about off-target drifts across the settings in Table 5.

Setting	Candidate (N_C)	Target (N_T)	Reference (N_R)	Base Model	LoRA rank	LoRA alpha	Rank $K_R + K_T$	(ρ, ϵ)	Entanglement ($\pi_{Q_{\text{clean}}}, \pi_{Q_{\text{ent}}}$)
1	CodeAlpaca (20K) [34]	HumanEval (164) [35]	StereoSet (4.2K) [36]	Base	16	16	512 + 128	$(1 \times 10^{-5}, 1)$	(0.16, 0.31)
2	MedMCQA (2K) [37]	MedQA (256) [38]	BBQ (253) [39] TruthfulQA (256) [40]	Base- Instruct	16	32	64 + 16	(1, 1)	(0.22, 0.77)
3	Empathetic Dialogues (5K) [41]	ESConv (100) [42]	GSM8K (256) [43]	Base- Instruct	8	16	64 + 16	(0.01, 1)	(0.36, 0.08)
4	FLAN (100K) [44]	IFEval (42) [45]	MMLU (285) [46]	Base	32	128	128 + 32	$(2 \times 10^{-5}, 2)$	(0.22, 0.26)

Table 1: *Summary of experimental settings. N_C : candidate pool size. N_T : target set size (used for $\hat{g}_{\text{tar}}$, validation, and test). N_R : reference set size (used for Fisher estimation and off-target drift evaluation). The definitions of $\pi_{Q_{\text{clean}}}$ and $\pi_{Q_{\text{ent}}}$ are defined and illustrated in Appendix D.2 and Remark 3. All runs use learning rate $\eta = 2 \times 10^{-4}$.*

3.1 Main Results

The settings in Table 1 fall into two regimes, depending on whether fine-tuning on the candidate pool degrades or improves model behavior on the reference task. Settings 1, 2, and 4 fall in the first, where target gains come at the cost of off-target drift. Setting 3 falls in the second: its entanglement score $\pi_{Q_{\text{ent}}} = 0.08$ (Table 1) is far below that of the other settings, so target-improving candidates lie largely outside the reference-sensitive directions and the target can be improved at zero off-target cost. Table 2 reports the target performance and the off-target behavioral drift D_{task} across all settings.

Case 1: fine-tuning degrades reference behavior (Settings 1, 2, and 4). In Setting 1, fine-tuning on CodeAlpaca improves Python programming ability but increases stereotype bias: all baselines increase the StereoSet score relative to the base model, where a smaller change indicates better preservation of the base model’s stereotype bias. SAFE-DRIFT achieves competitive pass@1 and pass@10 rates on HumanEval while producing the smallest $\Delta\text{StereoSet}$ score among all methods.

Setting 2 illustrates the trade-off more sharply. Although fine-tuning improves MedQA performance, reference metrics deteriorate for most baselines: accuracy on TruthfulQA and BBQ, which probe factual reliability and social-bias behavior, drops for random, DSIR, and LESS. SAFE-DRIFT sacrifices some MedQA accuracy in exchange for substantially smaller losses on off-target drift. The high entanglement score $\pi_{Q_{\text{ent}}} = 0.77$ indicates that most target-improving candidates are entangled with reference-sensitive directions, explaining why controlling drift necessarily constrains target performance in this setting.

Setting 4 presents a complementary picture. With $\pi_{Q_{\text{ent}}} = 0.26$ and $\pi_{Q_{\text{clean}}} = 0.22$, a larger share of candidates carry target signal in directions decoupled from the reference geometry. SAFE-DRIFT

²Although introduced for synthetic data generation, we adapt it for data selection, motivated by its use of gradient diversity to capture meaningful variation across data instances.

Setting	Metric	Random	DSIR	LESS	Prismatic	SAFE-DRIFT
1 (15%)	Δ HumanEval pass@1	0.136 $\pm$ 0.005	0.122 $\pm$ 0.015	0.116 $\pm$ 0.003	0.118 $\pm$ 0.016	<u>0.132 $\pm$ 0.004</u>
	Δ HumanEval pass@10	0.207 $\pm$ 0.023	0.150 $\pm$ 0.010	0.162 $\pm$ 0.017	0.175 $\pm$ 0.026	<u>0.183 $\pm$ 0.023</u>
	Δ StereoSet score	<u>0.537 $\pm$ 0.519</u>	1.162 $\pm$ 0.455	0.999 $\pm$ 0.706	1.012 $\pm$ 0.745	0.148 $\pm$ 0.662
2 (25%)	Δ MedQA acc.	<u>0.033 $\pm$ 0.004</u>	0.039 $\pm$ 0.016	0.021 $\pm$ 0.003	0.027 $\pm$ 0.013	0.015 $\pm$ 0.006
	Δ TruthfulQA acc.	-0.069 $\pm$ 0.023	-0.083 $\pm$ 0.019	<u>-0.062 $\pm$ 0.005</u>	-0.080 $\pm$ 0.022	-0.038 $\pm$ 0.010
	Δ BBQ acc.	-0.026 $\pm$ 0.036	-0.004 $\pm$ 0.011	0.014 $\pm$ 0.042	-0.071 $\pm$ 0.000	0.007 $\pm$ 0.033
3 (10%)	Δ ESCConv	<u>0.047 $\pm$ 0.080</u>	0.039 $\pm$ 0.007	0.053 $\pm$ 0.007	0.037 $\pm$ 0.008	0.039 $\pm$ 0.006
	Δ GSM8K acc.	0.064 $\pm$ 0.007	0.065 $\pm$ 0.004	<u>0.072 $\pm$ 0.008</u>	0.069 $\pm$ 0.015	0.084 $\pm$ 0.004
	Δ MMLU acc.	0.030 $\pm$ 0.007	0.025 $\pm$ 0.008	<u>0.039 $\pm$ 0.004</u>	0.029 $\pm$ 0.006	0.044 $\pm$ 0.010
4 (5%)	Δ IFEval prompt acc.	3.530 $\pm$ 2.435	-0.407 $\pm$ 0.665	2.172 $\pm$ 1.545	3.259 $\pm$ 1.090	4.141 $\pm$ 2.145
	Δ IFEval inst. acc.	<u>3.728 $\pm$ 1.886</u>	0.877 $\pm$ 1.236	2.632 $\pm$ 1.162	3.158 $\pm$ 1.058	4.649 $\pm$ 1.911
	Δ MMLU acc.	-0.833 $\pm$ 1.434	-0.667 $\pm$ 0.850	<u>0.500 $\pm$ 0.816</u>	-0.500 $\pm$ 2.828	0.500 $\pm$ 0.408

Table 2: Target gain and off-target drift D_{task} across all settings. All values are changes Δ relative to the base model. For each setting, target metrics (shaded) are separated by a horizontal rule from reference metrics (details in Appendix D). Bold denotes the best result per row; underline the second best. Scores are means $\pm$ standard deviation over three random seeds.

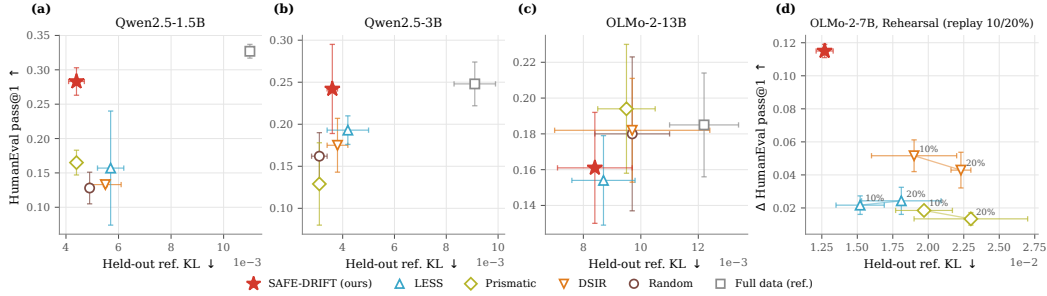


Figure 2: Realized drift D_{KL} vs. target gain (Setting 1). (a–c) Ablation across Qwen2.5 and OLMo-2 base models. (d) Comparison against rehearsal baselines on OLMo-2-7B: each baseline is fine-tuned with 10% or 20% of replayed reference data, whereas SAFE-DRIFT uses no replay. We plot the mean and standard deviation across three seeds; upper left is better on the Pareto plots.

achieves the best performance on both target metrics and improve MMLU accuracy, while all baselines except LESS degrade it, confirming that clean task-relevant signal allows the method to improve the target without incurring significant off-target drift.

Case 2: fine-tuning improves reference behavior (Setting 3). Setting 3 has by far the lowest entanglement of the four, $\pi_{Q_{\text{ent}}} = 0.08$ (Table 1): almost no candidate carries its target signal through reference-sensitive directions, so the drift budget is slack and SAFE-DRIFT can follow the target gradient nearly unconstrained. The reference metrics improve—emotional support dialogue draws on broad world knowledge and social reasoning, the same general capabilities probed by MMLU and GSM8K, so gains transfer rather than trade-off. SAFE-DRIFT matches DSIR and Prismatic on the target task while achieving the largest gains on both GSM8K and MMLU. Accordingly, selection is driven almost entirely by target alignment and the selected set includes points whose residual Fisher cost is not small

Comparison against behavior-preservation methods. We additionally compare against rehearsal approach applied to above baselines, the standard remedy for catastrophic forgetting (Setting 1, Figure 2d). To our knowledge SAFE-DRIFT is the first to formulate drift control from a reference Fisher perspective and to use it to select data under an explicit budget. Rehearsal mixes held-out reference examples into training at 10% and 20% of the dataset. SAFE-DRIFT attains the largest target gain compared to any rehearsal-based baselines, with the lowest realized drift, and without ever training on the reference set. Full numbers are in Appendix D.5.

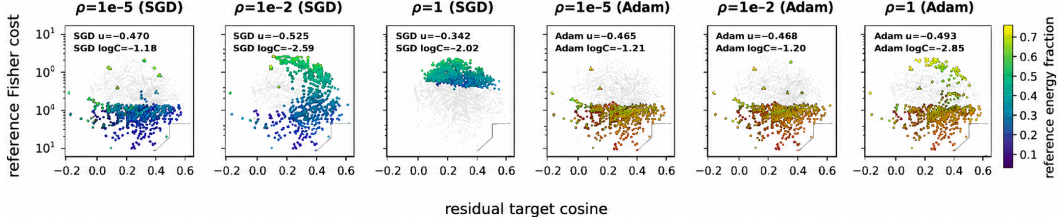


Figure 3: *Effect of the drift budget ρ on target accuracy and off-target drift under SGD and Adam preconditioners. Colored markers denote the examples selected by SAFE-DRIFT in Setting 2. What is u and $\log C$ here? Needs explanation.*

3.2 Ablation Studies

Ablation across drift budget ρ and gradient update methods. We sweep ρ over $\{10^{-5}, 10^{-2}, 10^0\}$ in Setting 2 (Figure 3). Under the SGD preconditioner ($\mathbf{P} = \mathbf{I}$), increasing ρ moves the selected set from mostly low-cost candidates toward a high-cost, high-entanglement region, consistent with $\pi_{Q_{\text{ent}}} = 0.77$; under Adam the selected set is nearly unchanged between $\rho = 10^{-5}$ and 10^{-2} and enters that region only at $\rho = 1$, so the preconditioner keeps the effective update in a low-cost region for longer. Across a larger sweep (Table 5), larger ρ is consistently associated with higher target accuracy, while its association with reference drift reverses sign between the entangled and aligned settings. We suggest reading $\sqrt{\rho}/2$ as a rough upper bound on the increase in probability of any undesirable output under the fine-tuned model, or select it on a validation drift set (Appendix D.7).

Robustness across model families and scales. We repeat Setting 1 across Qwen2.5 at 1.5B/3B/7B and OLMo-2 at 7B/13B, at a 15% budget with three seeds (Figure 2a–c; full table in Appendix D.6). SAFE-DRIFT attains the lowest predicted drift $D_{\mathcal{R}}$ in every configuration, reducing $D_{\mathcal{R}}$ by 10.2–18% and D_{KL} by 31.1–60% relative to full data at competitive target gain. [Add more details on comparison across scales.](#)

Computational cost and further ablations. SAFE-DRIFT shares its dominant cost—gradient construction over the candidate pool—with LESS and Prismatic, so its selection FLOPs are identical with other gradient-based selectors (Appendix D.11). Ablations on the selection budgets, rank K , the update geometry, preconditioners, mixture of reference tasks, and the LoRA versus full-SFT parameterization are provided in Appendix D.4–D.10. Our method is robust across configurations and extends to references composed of multiple benchmarks, consistently achieving the lowest drift on protected and unprotected tasks alike.

4 Conclusion

Data selection methods are typically evaluated on target task alone, without accounting for their impact on model behavior outside the target domain. In this paper, we explicitly formalize this trade-off between target improvement and off-target drift. Our method, SAFE-DRIFT, enables controllable off-target drift during targeted data selection. Through systematic experiments across various settings and domains, we demonstrate the effectiveness of our method. However, our method relies on two assumptions. First, it requires a well-approximated reference distribution. If the reference set does not fully cover all the behaviors we want to preserve, such as reasoning ability or domain knowledge, the observed reference distribution may underestimate the true off-target drift. We believe a future direction is to study how reference distribution should be constructed, expanded, and adapted rather than treated as fixed inputs, for example, following related work on broad-coverage evaluation, red-teaming evaluation, and trustworthiness [47–50]. Second, our analysis is based on a one-step or local update approximation which is a reasonable assumption for supervised fine-tuning, but it might not hold for fine-tuning based on reinforcement learning, where parameter updates can grow large as the policy diverges from initialization. We encourage future work to handle the trade-off in such settings where drift accumulates across updates.

Acknowledgments

C.D. would like to acknowledge support for this project from the National Science Foundation (NSF grant IIS-2238616). This work was completed in part with resources provided by the University of Chicago’s Research Computing Center.

References

- [1] Xinlu Zhang, Chenxin Tian, Xianjun Yang, Lichang Chen, Zekun Li, and Linda Ruth Petzold. Alpacare: Instruction-tuned large language models for medical application. *arXiv preprint arXiv:2310.14558*, 2023.
- [2] Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.
- [3] Pierre Colombo, Telmo Pires, Malik Boudiaf, Rui Melo, Dominic Culver, Etienne Malaboeuf, Gabriel Hautreux, Johanne Charpentier, and Michael Desa. Saullm-54b & saullm-141b: Scaling up domain adaptation for the legal domain. *Advances in Neural Information Processing Systems*, 37:129672–129695, 2024.
- [4] Joel Niklaus, Lucia Zheng, Arya D McCarthy, Christopher Hahn, Brian M Rosen, Peter Henderson, Daniel E Ho, Garrett Honke, Percy Liang, and Christopher Manning. Flawn-t5: An empirical examination of effective instruction-tuning data mixtures for legal reasoning. *arXiv preprint arXiv:2404.02127*, 2024.
- [5] Lucia Zheng, Neel Guha, Brandon R Anderson, Peter Henderson, and Daniel E Ho. When does pretraining help? assessing self-supervised learning for law and the casehold dataset of 53,000+ legal holdings. In *Proceedings of the eighteenth international conference on artificial intelligence and law*, pages 159–168, 2021.
- [6] Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024.
- [7] Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody Blakeney, and John P. Cunningham. Lora learns less and forgets less, 2024. URL <https://arxiv.org/abs/2405.09673>.
- [8] Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, LILI YU, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. LIMA: Less is more for alignment. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=KBMOkmX2he>.
- [9] Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. Limo: Less is more for reasoning, 2025. URL <https://arxiv.org/abs/2502.03387>.
- [10] Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Guoyin Wang, et al. Instruction tuning for large language models: A survey. *ACM Computing Surveys*, 58(7):1–36, 2026.
- [11] Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning aligned language models compromises safety, even when users do not intend to! *arXiv preprint arXiv:2310.03693*, 2023.
- [12] Jan Betley, Niels Warncke, Anna Sztyber-Betley, Daniel Tan, Xuchan Bao, Martín Soto, Megha Srivastava, Nathan Labenz, and Owain Evans. Training large language models on narrow tasks can lead to broad misalignment. *Nature*, 649(8097):584–589, 2026.
- [13] Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *IEEE Transactions on Audio, Speech and Language Processing*, 2025.

- [14] Yihan Cao, Yanbin Kang, Chi Wang, and Lichao Sun. Instruction mining: Instruction data selection for tuning large language models. *arXiv preprint arXiv:2307.06290*, 2023.
- [15] Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. Ccnet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the twelfth language resources and evaluation conference*, pages 4003–4012, 2020.
- [16] Alexander Wettig, Aatmik Gupta, Saumya Malik, and Danqi Chen. Qurating: Selecting high-quality data for training language models. *arXiv preprint arXiv:2402.09739*, 2024.
- [17] Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. Data selection for language models via importance resampling, 2023. URL <https://arxiv.org/abs/2302.03169>.
- [18] Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Understanding predictions with data and data with predictions. In *International Conference on Machine Learning*, pages 9525–9587. PMLR, 2022.
- [19] Jaehun Jung, Seungju Han, Ximing Lu, Skyler Hallinan, David Acuna, Shrimai Prabhumoye, Mostofa Patwary, Mohammad Shoeybi, Bryan Catanzaro, and Yejin Choi. Prismatic synthesis: Gradient-based data diversification boosts generalization in llm reasoning. *arXiv preprint arXiv:2505.20161*, 2025.
- [20] Xu Yang, Chen Liu, and Ying Wei. Data selection matters: Towards robust instruction tuning of large multimodal models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.
- [21] 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. *arXiv preprint arXiv:2402.16827*, 2024.
- [22] Kiho Park, Todd Nief, Yo Joong Choe, and Victor Veitch. The information geometry of softmax: Probing and steering. *arXiv preprint arXiv:2602.15293*, 2026.
- [23] Vincent Siu, Nicholas Crispino, David Park, Nathan W Henry, Zhun Wang, Yang Liu, Dawn Song, and Chenguang Wang. Steeringsafety: A systematic safety evaluation framework of representation steering in llms. *arXiv preprint arXiv:2509.13450*, 2025.
- [24] Jinman Wu, Yi Xie, Shen Lin, Shiqian Zhao, and Xiaofeng Chen. Knowing without acting: The disentangled geometry of safety mechanisms in large language models. *arXiv preprint arXiv:2603.05773*, 2026.
- [25] Austin C Kozlowski, Callin Dai, and Andrei Boutyline. Semantic structure in large language model embeddings. *arXiv preprint arXiv:2508.10003*, 2025.
- [26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [27] Stéphane G Mallat and Zhifeng Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on signal processing*, 41(12):3397–3415, 1993.
- [28] Krishnateja Killamsetty, Sivasubramanian Durga, Ganesh Ramakrishnan, Abir De, and Rishabh Iyer. Grad-match: Gradient matching based data subset selection for efficient deep model training. In *International Conference on Machine Learning*, pages 5464–5474. PMLR, 2021.
- [29] Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (volume 1: long papers)*, pages 7319–7328, 2021.
- [30] 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.

- [31] Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. Galore: Memory-efficient llm training by gradient low-rank projection. *arXiv preprint arXiv:2403.03507*, 2024.
- [32] Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning, 2025. URL <https://arxiv.org/abs/2308.08747>.
- [33] Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Allyson Ettinger, Michal Guerquin, David Heineman, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Jake Poznanski, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 olmo 2 furious, 2025. URL <https://arxiv.org/abs/2501.00656>.
- [34] Sahil Chaudhary. Code alpaca: An instruction-following llama model for code generation. <https://github.com/sahil280114/codealpaca>, 2023.
- [35] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. 2021.
- [36] Moin Nadeem, Anna Bethke, and Siva Reddy. Stereoset: Measuring stereotypical bias in pretrained language models, 2020.
- [37] Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In Gerardo Flores, George H Chen, Tom Pollard, Joyce C Ho, and Tristan Naumann, editors, *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pages 248–260. PMLR, 07–08 Apr 2022. URL <https://proceedings.mlr.press/v174/pal22a.html>.
- [38] Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021.
- [39] Alicia Parrish, Angelica Chen, Nikita Nangia, Vishakh Padmakumar, Jason Phang, Jana Thompson, Phu Mon Htut, and Samuel Bowman. BBQ: A hand-built bias benchmark for question answering. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2086–2105, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.165. URL <https://aclanthology.org/2022.findings-acl.165/>.
- [40] Harsh Raj, Domenic Rosati, and Subhabrata Majumdar. Measuring reliability of large language models through semantic consistency. *arXiv preprint arXiv:2211.05853*, 2022.
- [41] Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Annual Meeting of the Association for Computational Linguistics*, 2018. URL <https://api.semanticscholar.org/CorpusID:195069365>.

- [42] Siyang Liu, Chujie Zheng, Orianna Demasi, Sahand Sabour, Yu Li, Zhou Yu, Yong Jiang, and Minlie Huang. Towards emotional support dialog systems. In *ACL*, 2021.
- [43] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.
- [44] Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Chandu, David Wadden, Kelsey MacMillan, Noah A Smith, Iz Beltagy, et al. How far can camels go? exploring the state of instruction tuning on open resources. *Advances in Neural Information Processing Systems*, 36:74764–74786, 2023.
- [45] Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models, 2023. URL <https://arxiv.org/abs/2311.07911>.
- [46] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.
- [47] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.
- [48] Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu, Zidi Xiong, Ritik Dutta, Rylan Schaeffer, et al. Decodingtrust: A comprehensive assessment of trustworthiness in gpt models. *Advances in Neural Information Processing Systems*, 36: 31232–31339, 2023.
- [49] Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, et al. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal. *arXiv preprint arXiv:2402.04249*, 2024.
- [50] Hwan Chang and Hwanhee Lee. Which retain set matters for llm unlearning? a case study on entity unlearning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5966–5982, 2025.
- [51] Mayee F Chen, Nicholas Roberts, Kush Bhatia, Jue WANG, Ce Zhang, Frederic Sala, and Christopher Re. Skill-it! a data-driven skills framework for understanding and training language models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=Ioizw01Nlf>.
- [52] Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. When less is more: Investigating data pruning for pretraining llms at scale, 2023. URL <https://arxiv.org/abs/2309.04564>.
- [53] Hamish Ivison, Noah A. Smith, Hannaneh Hajishirzi, and Pradeep Dasigi. Data-efficient finetuning using cross-task nearest neighbors. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9036–9061, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.576. URL <https://aclanthology.org/2023.findings-acl.576/>.
- [54] Sang Keun Choe, Hwijee Ahn, Juhan Bae, Kewen Zhao, Youngseog Chung, Adithya Pratapa, Willie Neiswanger, Emma Strubell, Teruko Mitamura, Jeff Schneider, Eduard Hovy, Roger Baker Grosse, and Eric P. Xing. What is your data worth to GPT? LLM-scale data valuation with influence functions. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL <https://openreview.net/forum?id=zPKeJAEo27>.
- [55] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach, 2018. URL <https://arxiv.org/abs/1708.00489>.

- [56] Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models, 2020. URL <https://arxiv.org/abs/1906.01827>.
- [57] Ofer Yehuda, Avihu Dekel, Guy Hacohen, and Daphna Weinshall. Active learning through a covering lens, 2022. URL <https://arxiv.org/abs/2205.11320>.
- [58] Gantavya Bhatt, Yifang Chen, Arnav Das, Jifan Zhang, Sang Truong, Stephen Mussmann, Yinglun Zhu, Jeff Bilmes, Simon Du, Kevin Jamieson, Jordan Ash, and Robert Nowak. An experimental design framework for label-efficient supervised finetuning of large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6549–6560, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.390. URL <https://aclanthology.org/2024.findings-acl.390/>.
- [59] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE transactions on pattern analysis and machine intelligence*, 46(8):5362–5383, 2024.
- [60] Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. Orthogonal subspace learning for language model continual learning. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10658–10671, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.715. URL <https://aclanthology.org/2023.findings-emnlp.715/>.
- [61] Hippolyt Ritter, Aleksandar Botev, and David Barber. Online structured laplace approximations for overcoming catastrophic forgetting, 2018. URL <https://arxiv.org/abs/1805.07810>.
- [62] Ashwinee Panda, Berivan Isik, Xiangyu Qi, Sanmi Koyejo, Tsachy Weissman, and Prateek Mittal. Lottery ticket adaptation: Mitigating destructive interference in llms, 2024. URL <https://arxiv.org/abs/2406.16797>.
- [63] Frederik Kunstner, Philipp Hennig, and Lukas Balles. Limitations of the empirical fisher approximation for natural gradient descent. *Advances in neural information processing systems*, 32, 2019.
- [64] Sylvain Gugger, Lysandre Debut, Thomas Wolf, Philipp Schmid, Zachary Mueller, Sourab Mangrulkar, Marc Sun, and Benjamin Bossan. Accelerate: Training and inference at scale made simple, efficient and adaptable. <https://github.com/huggingface/accelerate>, 2022.
- [65] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language model with self generated instructions, 2022.
- [66] Yuling Gu, Oyvind Tafjord, Bailey Kuehl, Dany Haddad, Jesse Dodge, and Hannaneh Hajishirzi. Olmes: A standard for language model evaluations, 2025. URL <https://arxiv.org/abs/2406.08446>.

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: Our abstract accurately reflects all the findings presented in the paper

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: Limitations are discussed in the conclusion section.

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: We provide the detailed proof in the corresponding section as well as extended proof in the appendix.

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: The appendix sections contain the necessary hyperparameters and implementation details to reproduce our work. Furthermore, we provide code to reproduce the results and figures.

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 as a supplemental file the code necessary to reproduce all results.

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: The results and appendix provides all necessary experimental details to fully understand and interpret our results.

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: [Yes]

Justification: We use error bars for all the reported results.

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: We provide information on the compute resources used in the appendix.

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: Our research fully conforms to the code of ethics.

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: This paper provides new tools for safeguarding large language models against unwanted behavioral drift during fine-tuning, helping preserve important capabilities and reduce unintended misalignment.

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: There are no risks associated with our paper or code.

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 properly attribute all relevant software and datasets used.

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: The new asset introduced in this paper is the code implementation of our SAFE-DRIFT algorithm used in the experiments. We plan to release the code with accompanying documentation, including instructions for setup, preprocessing, running experiments, and reproducing the main results.

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: There were no human subjects involved in our research.

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: There were no human subjects involved in our research.

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: The paper does not involve the use of large language models (LLMs) as an important, original, or non-standard component of the core methods.

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.

A Related Work & Notations

A.1 Related Works

Data Selection Data selection aims to find a subset of training examples that induces performance similar to training on the full dataset. In particular, [21] divide data selection methods in two main classes: Distribution Matching methods, and Distribution Diversification methods.

Distribution matching methods aim to find samples that closely match the target distribution. In this setting, a crucial component is to assess the similarity of a given data point with the target data. A plethora of methods have introduced non-model specific notions of useful data [51] or n -gram features [17] to perform data selection for pre-training. Because these methods lack the finesse of detecting “high-quality” data points by solely relying on surface attributes, the general trend in the literature – particularly in the context of fine-tuning or instruction tuning — seems to have turned towards using model-specific metrics based on scores like perplexity [52], model embedding [53]. Recent methods have for instance relied on formulating the data selection problem as a multi-armed bandit, where the utility function is an aggregated reward function of the cosine similarity and magnitude similarity between gradients of batches from the auxiliary and target datasets. Another line of recent work relies on model gradients [6, 54] for data selection, and calculate the pointwise data influence between pair of training and validation instances.

Distribution diversification methods, by contrast, seek to emphasize diversity of instances. These algorithms aim to efficiently approximate various geometric extent measures over the training set using a small subset. These include coresets methods [55, 56] and submodular algorithms [57, 58].

Catastrophic Forgetting Catastrophic forgetting [59, 32] happens when adaptation to a new distribution generally results in a largely reduced ability to capture the old ones. To address this issue, Existing approaches to remedy this issue can be typically divided into two categories: data-aware and data-oblivious. Data-aware methods [60, 61] assume access to earlier data distributions to some extent. In contrast, data-oblivious methods operate solely with the pre-trained model, without revisiting any prior data. Data-oblivious methods assume access to the pretrained model without revisiting any prior data. Previous works [62, 7] selectively update a subset of parameters in a pretrained model to limit interference with previously knowledge.

A.2 Notations.

Throughout the paper, we use bold uppercase letters, such as $\mathbf{A}$, to denote matrices. For a matrix $\mathbf{A} \in \mathbb{R}^{d \times n}$, we write $\mathbf{A}(i)$ for its i -th column and $\mathbf{A}_{ij}$ for its (i, j) -th entry. For a vector $a \in \mathbb{R}^d$, we write a_i for its i -th coordinate. Let $\text{diag}(\mathbf{A}, \mathbf{B})$ denote the block-diagonal matrix $\begin{bmatrix} \mathbf{A} & 0 \\ 0 & \mathbf{B} \end{bmatrix}$ given matrices $\mathbf{A}$ and $\mathbf{B}$. Let $\mathbf{0}_K \in \mathbb{R}^{K \times K}$ denote the zero matrix, $\mathbf{I}_K \in \mathbb{R}^{K \times K}$ denote the identity matrix for some constant $K > 0$.

B Technical Appendices

B.1 Proof of Proposition 1

Proof. Assume there exists $\rho > 0$ such that the following hold in the neighborhood $\mathcal{N}_\rho(\theta_0) := \{\theta \in \mathbb{R}^d : \|\theta - \theta_0\|_2 < \rho\}$:

- (i) For all (x, y) in the support of the joint distribution $x \sim \mathcal{R}$ and $y \sim p_{\theta_0}(\cdot | x)$, the map $\theta \mapsto \log p_\theta(y | x)$ is three times continuously differentiable in $\mathcal{N}_\rho(\theta_0)$.
- (ii) There exists an integrable envelope $G(x, y)$ such that

$$\sup_{\theta \in \mathcal{N}_\rho(\theta_0)} \|\nabla_\theta^3 \log p_\theta(y | x)\| \leq G(x, y), \quad \mathbb{E}_{x \sim \mathcal{R}} \mathbb{E}_{y \sim p_{\theta_0}(\cdot | x)} [G(x, y)] < \infty.$$

For a fixed x , define

$$d_x(\theta) = D_{\text{KL}}(p_{\theta_0}(\cdot | x) \| p_\theta(\cdot | x)).$$

Then,

$$D_{\mathcal{R}}(\Delta\theta) = \mathbb{E}_{x \sim \mathcal{R}}[d_x(\theta_0 + \Delta\theta)].$$

From the definition of KL divergence,

$$d_x(\theta) = \mathbb{E}_{p_{\theta_0}(\cdot|x)}[\log p_{\theta_0}(\cdot|x)] - \mathbb{E}_{p_{\theta_0}(\cdot|x)}[\log p_{\theta}(\cdot|x)]$$

and

$$\nabla_{\theta} d_x(\theta) = \mathbb{E}_{p_{\theta_0}(\cdot|x)}[-\nabla_{\theta} \log p_{\theta}(\cdot|x)], \quad \nabla_{\theta}^2 d_x(\theta) = \mathbb{E}_{p_{\theta_0}(\cdot|x)}[-\nabla_{\theta}^2 \log p_{\theta}(\cdot|x)].$$

At $\theta = \theta_0$,

$$\begin{aligned} \nabla_{\theta} d_x(\theta)|_{\theta=\theta_0} &= \mathbb{E}_{p_{\theta_0}(\cdot|x)}[-\nabla_{\theta} \log p_{\theta_0}(\cdot|x)] \\ &= \int p_{\theta_0}(\cdot|x) \frac{\nabla_{\theta} p_{\theta_0}(\cdot|x)}{p_{\theta_0}(\cdot|x)} \\ &= \nabla_{\theta} \int p_{\theta_0}(\cdot|x) \\ &= \mathbf{0}_d \end{aligned}$$

and

$$\nabla_{\theta}^2 d_x(\theta)|_{\theta=\theta_0} = \mathbb{E}_{p_{\theta_0}(\cdot|x)}[-\nabla_{\theta}^2 \log p_{\theta_0}(\cdot|x)].$$

By a second-order Taylor expansion around $\theta = \theta_0$,

$$\begin{aligned} d_x(\theta) &= d_x(\theta_0) + \nabla_{\theta} d_x(\theta)|_{\theta=\theta_0}^{\top}(\theta - \theta_0) + \frac{1}{2}(\theta - \theta_0)^{\top} \nabla_{\theta}^2 d_x(\theta)|_{\theta=\theta_0}(\theta - \theta_0) + R_x(\Delta\theta) \\ &= \frac{1}{2} \Delta\theta^{\top} \mathbb{E}_{p_{\theta_0}(\cdot|x)}[-\nabla_{\theta}^2 \log p_{\theta_0}(\cdot|x)] \Delta\theta + R_x(\Delta\theta), \end{aligned}$$

where $R_x(\Delta\theta)$ is Taylor remainder. We combine with the result from [63],

$$\mathbf{F}_{\mathcal{R}} = \mathbb{E}_{x \sim \mathcal{R}} \mathbb{E}_{y \sim p_{\theta_0}(\cdot|x)}[-\nabla_{\theta}^2 \log p_{\theta_0}(\cdot|x)],$$

which leads to,

$$D_{\mathcal{R}}(\Delta\theta) = \frac{1}{2} \Delta\theta^{\top} \mathbf{F}_{\mathcal{R}} \Delta\theta + \mathbb{E}_{x \sim \mathcal{R}}[R_x(\Delta\theta)].$$

By Taylor's theorem, for each fixed x and y there exists $\tilde{\theta} \in [\theta_0, \theta_0 + \Delta\theta]$ such that the third-order remainder satisfies

$$|R_x(\Delta\theta)| \leq \frac{1}{6} \sup_{\theta \in \mathcal{N}_{\rho}(\theta_0)} \|\nabla_{\theta}^3 \log p_{\theta}(y|x)\| \cdot \|\Delta\theta\|_2^3.$$

Using the envelope assumption, we obtain

$$|R_x(\Delta\theta)| \leq \frac{1}{6} G(x, y) \|\Delta\theta\|_2^3.$$

Taking expectation over $x \sim \mathcal{R}$ and $y \sim p_{\theta_0}(\cdot|x)$ gives

$$\left| D_{\mathcal{R}}(\Delta\theta) - \frac{1}{2} \Delta\theta^{\top} \mathbf{F}_{\mathcal{R}} \Delta\theta \right| \leq \frac{\|\Delta\theta\|_2^3}{6} \mathbb{E}_{x \sim \mathcal{R}} \mathbb{E}_{y \sim p_{\theta_0}(\cdot|x)}[G(x, y)].$$

Since the envelope is integrable,

$$D_{\mathcal{R}}(\Delta\theta) = \frac{1}{2} \Delta\theta^{\top} \mathbf{F}_{\mathcal{R}} \Delta\theta + o(\|\Delta\theta\|_2^2).$$

This proves the claim. $\square$

B.2 Proof of Proposition 2

Theorem 1 (Oracle safe update (full)). *Let $\mathbf{F}_{\mathcal{R}} \succeq 0$ and let $g_{\mathcal{T}} \neq 0$. For $\rho > 0$ and $\varepsilon > 0$, consider*

$$\Delta^* \in \arg \min_{\Delta \in \mathbb{R}^d} g_{\mathcal{T}}^\top \Delta \quad \text{subject to} \quad \frac{1}{2} \Delta^\top \mathbf{F}_{\mathcal{R}} \Delta \leq \rho, \quad \|\Delta\|_2 \leq \varepsilon.$$

Then Δ^ is characterized as follows.*

1. **Loose drift constraint.** *If*

$$\frac{\varepsilon^2}{2\|g_{\mathcal{T}}\|_2^2} g_{\mathcal{T}}^\top \mathbf{F}_{\mathcal{R}} g_{\mathcal{T}} \leq \rho,$$

then the reference-drift constraint is inactive and

$$\Delta^* = -\varepsilon \frac{g_{\mathcal{T}}}{\|g_{\mathcal{T}}\|_2}.$$

2. **Loose norm constraint.** *Suppose $g_{\mathcal{T}} \in \text{Range}(\mathbf{F}_{\mathcal{R}})$ and define*

$$\Delta_R = -\sqrt{\frac{2\rho}{g_{\mathcal{T}}^\top \mathbf{F}_{\mathcal{R}}^\dagger g_{\mathcal{T}}}} \mathbf{F}_{\mathcal{R}}^\dagger g_{\mathcal{T}},$$

where $\mathbf{F}_{\mathcal{R}}^\dagger$ denotes the Moore–Penrose pseudoinverse. If

$$\|\Delta_R\|_2 \leq \varepsilon,$$

then the norm constraint is inactive and

$$\Delta^* = \Delta_R.$$

3. **Both constraints active.** *In all remaining nondegenerate cases, both constraints are active. There exists a unique damping parameter $\alpha > 0$ such that*

$$\frac{\varepsilon^2}{2} \frac{g_{\mathcal{T}}^\top (\mathbf{F}_{\mathcal{R}} + \alpha I)^{-1} \mathbf{F}_{\mathcal{R}} (\mathbf{F}_{\mathcal{R}} + \alpha I)^{-1} g_{\mathcal{T}}}{g_{\mathcal{T}}^\top (\mathbf{F}_{\mathcal{R}} + \alpha I)^{-2} g_{\mathcal{T}}} = \rho,$$

and the solution is

$$\Delta^* = -\varepsilon \frac{(\mathbf{F}_{\mathcal{R}} + \alpha I)^{-1} g_{\mathcal{T}}}{\|(\mathbf{F}_{\mathcal{R}} + \alpha I)^{-1} g_{\mathcal{T}}\|_2}.$$

Equivalently, every nonzero solution satisfies the KKT representation

$$\Delta^* = -c (\mathbf{F}_{\mathcal{R}} + \alpha I)^{-1} g_{\mathcal{T}}$$

for some $c > 0$ and $\alpha \geq 0$, with the conventions that $\alpha = \infty$ corresponds to the Euclidean gradient-aligned solution in case 1, and $\alpha = 0$ corresponds to the reference-natural-gradient solution in case 2.

Proof. Rewrite (6) as a minimization:

$$\min_{\Delta\theta \in \mathbb{R}^d} g_{\mathcal{T}}^\top \Delta\theta \quad \text{subject to} \quad \frac{1}{2} \Delta\theta^\top \mathbf{F}_{\mathcal{R}} \Delta\theta \leq \rho, \quad \|\Delta\theta\|_2^2 \leq \varepsilon^2. \quad (13)$$

The max objective is linear and both constraints are convex since $\mathbf{F}_{\mathcal{R}} \succeq 0$ and $\|\cdot\|_2$ is convex. Note here that $\mathbf{F}_{\mathcal{R}} \succeq 0$ always holds since for any $v \in \mathbb{R}^d$, $v^\top \mathbf{F}_{\mathcal{R}} v = \mathbb{E}[(v^\top s_{\theta_0}(x, y))^2] \geq 0$. Taking $\Delta\theta = 0$ gives $0 \leq \rho$ and $0 \leq \varepsilon$, which is strictly feasible for any $\rho, \varepsilon > 0$. Slater's condition and strong duality hold. The KKT conditions are necessary and sufficient for a global optimum.

Form the Lagrangian with multipliers $\sigma \geq 0$ and $\mu \geq 0$:

$$\mathcal{L}(\Delta\theta, \sigma, \mu) = g_{\mathcal{T}}^\top \Delta\theta + \sigma \left(\frac{1}{2} \Delta\theta^\top \mathbf{F}_{\mathcal{R}} \Delta\theta - \rho \right) + \mu (\|\Delta\theta\|_2^2 - \varepsilon^2).$$

Stationarity. Setting $\nabla_{\Delta\theta}\mathcal{L} = 0$:

$$g_{\mathcal{T}} + \sigma \mathbf{F}_{\mathcal{R}} \Delta\theta + \mu \Delta\theta = 0 \implies (\sigma \mathbf{F}_{\mathcal{R}} + 2\mu \mathbf{I}) \Delta\theta = -g_{\mathcal{T}}. \quad (14)$$

For any $\mu > 0$, the matrix $\sigma \mathbf{F}_{\mathcal{R}} + 2\mu \mathbf{I}$ has an inverse regardless of whether $\mathbf{F}_{\mathcal{R}}$ is singular, and:

$$\Delta\theta^* = -(\sigma \mathbf{F}_{\mathcal{R}} + 2\mu \mathbf{I})^{-1} g_{\mathcal{T}}. \quad (15)$$

Complementary slackness.

$$\sigma \left(\frac{1}{2} (\Delta\theta^*)^\top \mathbf{F}_{\mathcal{R}} \Delta\theta^* - \rho \right) = 0, \quad (16)$$

$$\mu (\|\Delta\theta^*\|_2^2 - \epsilon^2) = 0. \quad (17)$$

We consider the three non-trivial cases.

Case 1: $\sigma = 0, \mu = 0$. Stationarity gives $g_{\mathcal{T}} = 0$, which is trivial. Assume $g_{\mathcal{T}} \neq 0$, so this case does not arise.

Case 2: $\sigma > 0, \mu = 0$ (**only KL constraint active**). By (17), the norm constraint is inactive. By (16), the KL constraint is tight:

$$\frac{1}{2} (\Delta\theta^*)^\top \mathbf{F}_{\mathcal{R}} \Delta\theta^* = \rho.$$

Substituting (15) with $\mu = 0$:

$$\frac{1}{2\sigma^2} g_{\mathcal{T}}^\top \mathbf{F}_{\mathcal{R}}^\dagger \mathbf{F}_{\mathcal{R}} \mathbf{F}_{\mathcal{R}}^\dagger g_{\mathcal{T}} = \frac{1}{2\sigma^2} g_{\mathcal{T}}^\top \mathbf{F}_{\mathcal{R}}^\dagger g_{\mathcal{T}} = \rho,$$

giving $\sigma^* = \sqrt{g_{\mathcal{T}}^\top \mathbf{F}_{\mathcal{R}}^\dagger g_{\mathcal{T}} / (2\rho)}$, and the solution reduces to:

$$\Delta\theta^* = -\sqrt{\frac{2\rho}{g_{\mathcal{T}}^\top \mathbf{F}_{\mathcal{R}}^\dagger g_{\mathcal{T}}}} \mathbf{F}_{\mathcal{R}}^\dagger g_{\mathcal{T}}. \quad (18)$$

where $\mathbf{F}_{\mathcal{R}}^\dagger$ is the Moore–Penrose pseudoinverse. Let the decomposition of $g_{\mathcal{T}}$

$$g_{\mathcal{T}} = g_{\text{range}} + g_{\text{null}}, \quad g_{\text{null}} \in \text{Null}(\mathbf{F}_{\mathcal{R}}),$$

We note that the solution (18) is valid only when $g_{\text{null}} = 0$. If there is any direction $v \in \text{Null}(\mathbf{F}_{\mathcal{R}})$ with $g_{\mathcal{T}}^\top v < 0$, moving along v decreases the objective without changing the off-target cost, making the optimization unbounded. Thus, if $g_{\text{null}} \neq 0$, the norm constraint on $\|\Delta\theta\|_2$ must be active, so one must have $\mu > 0$ and the optimal solution cannot be achieved when $\mu = 0$.

Case 3: $\sigma = 0, \mu > 0$ (**norm constraint active**). By the slackness condition (17),

$$\|\Delta\theta^*\|_2^2 = \frac{1}{4\mu^2} \|g_{\mathcal{T}}\|_2^2 = \epsilon^2$$

leading to $\mu^* = \|g_{\mathcal{T}}\|_2 / 2\epsilon$. Thus,

$$\Delta\theta^* = -\frac{\epsilon}{\|g_{\mathcal{T}}\|_2} g_{\mathcal{T}}$$

When only the norm constraint is active, the optimizer ignores the off-target drift and reduces to a target gradient-aligned step.

Case 4: $\sigma > 0, \mu > 0$ (**both constraints active**). In all cases the solution takes the form (15), which we write compactly:

$$\Delta\theta^* = -\frac{1}{\sigma^*} (\mathbf{F}_{\mathcal{R}} + \alpha \mathbf{I})^{-1} g_{\mathcal{T}},$$

where $\sigma^* > 0$ scales the overall step size and is determined by whichever constraint is active, and $\alpha \geq 0$ controls the damping of $\mathbf{F}_{\mathcal{R}}$ by the norm constraint. When the both constraints are active, the solution is a damped natural gradient with damping coefficient determined by the ratio of the two Lagrange multipliers.

Using the active norm constraint ($\|\Delta\theta^*\|_2 = \epsilon$, we let $\sigma^* = \frac{\|(\mathbf{F}_{\mathcal{R}} + \alpha\mathbf{I})^{-1}g_{\mathcal{T}}\|_2}{\epsilon}$), and $\Delta\theta^*$ becomes

$$\Delta\theta^* = -\epsilon \frac{(\mathbf{F}_{\mathcal{R}} + \alpha\mathbf{I})^{-1}g_{\mathcal{T}}}{\|(\mathbf{F}_{\mathcal{R}} + \alpha\mathbf{I})^{-1}g_{\mathcal{T}}\|_2}, \quad (19)$$

Then, α is chosen to satisfy (16)

$$\frac{1}{2(\sigma^*)^2} g_{\mathcal{T}}^\top (\mathbf{F}_{\mathcal{R}} + \alpha\mathbf{I})^{-1} \mathbf{F}_{\mathcal{R}} (\mathbf{F}_{\mathcal{R}} + \alpha\mathbf{I})^{-1} g_{\mathcal{T}} = \rho \quad (20)$$

□

B.3 Characterization of α

Let $\mathbf{F}_{\mathcal{R}} = \mathbf{U}_{\mathcal{R}} \mathbf{\Lambda}_{\mathcal{R}} \mathbf{U}_{\mathcal{R}}^\top$ with eigenvalues λ_i and write the target gradient as $g_{\mathcal{T}} = \sum_i c_i v_i$ in the same basis. The oracle safe direction is then solved in Proposition 2. Equivalently, α is the positive root of the scalar equation $\phi(\alpha) = 2\rho/\epsilon^2$, where

$$\phi(\alpha) := \frac{\sum_i \frac{\lambda_i c_i^2}{(\lambda_i + \alpha)^2}}{\sum_i \frac{c_i^2}{(\lambda_i + \alpha)^2}}.$$

Proposition 3. *Let ϕ be defined as above. Then ϕ is nondecreasing on $(0, \infty)$. If there exist $i \neq j$ such that $c_i c_j \neq 0$ and $\lambda_i \neq \lambda_j$, then $\phi'(\alpha) > 0$ for all $\alpha > 0$. Consequently, whenever a positive solution exists, it is unique.*

Proof.

$$N(\alpha) := \sum_i \frac{\lambda_i c_i^2}{(\lambda_i + \alpha)^2}, \quad D(\alpha) := \sum_i \frac{c_i^2}{(\lambda_i + \alpha)^2}, \quad \phi(\alpha) = \frac{N(\alpha)}{D(\alpha)}.$$

Since $D(\alpha) > 0$ for all $\alpha > 0$, the quotient is well defined. We then differentiate both $N(\alpha)$ and $D(\alpha)$.

$$\phi'(\alpha) = \frac{N'(\alpha)D(\alpha) - N(\alpha)D'(\alpha)}{D(\alpha)^2}.$$

Moreover,

$$N'(\alpha) = -2 \sum_i \frac{\lambda_i c_i^2}{(\lambda_i + \alpha)^3}, \quad D'(\alpha) = -2 \sum_i \frac{c_i^2}{(\lambda_i + \alpha)^3},$$

so

$$N'(\alpha)D(\alpha) - N(\alpha)D'(\alpha) = 2 \sum_{i,j} \frac{\lambda_i c_i^2 c_j^2 (\lambda_i - \lambda_j)}{(\lambda_i + \alpha)^3 (\lambda_j + \alpha)^3}.$$

Pairing the (i, j) and (j, i) terms gives

$$N'(\alpha)D(\alpha) - N(\alpha)D'(\alpha) = \sum_{i < j} \frac{2c_i^2 c_j^2 (\lambda_i - \lambda_j)^2}{(\lambda_i + \alpha)^3 (\lambda_j + \alpha)^3} \geq 0.$$

Hence $\phi'(\alpha) \geq 0$ for all $\alpha > 0$. If there exist $i \neq j$ such that $c_i c_j \neq 0$ and $\lambda_i \neq \lambda_j$, then the corresponding pairwise term is strictly positive for every $\alpha > 0$, so $\phi'(\alpha) > 0$. □

Once the spectrum of $\mathbf{F}_{\mathcal{R}}$ and the target coefficients are known, α is determined by a one-dimensional monotone solve. See the Algorithm 2 for details on how to solve for α from (ρ, ϵ) in practice.

B.4 Choices of the Preconditioner $\mathbf{P}$

The influence function characterizes the implicit parameter change from re-optimization. In practice, however, fine-tuning uses explicit parameter update algorithms that approximate this implicit change. We unify these as:

$$\Delta\theta(w) = -\eta \mathbf{P} \mathbf{G} w,$$

where $\mathbf{P} \in \mathbb{R}^{d \times d}$ is a *preconditioner* encoding the geometry of the update algorithm, and $w \in \mathbb{R}_{\geq 0}^{N_C}$ is a non-negative weighting over training examples, and $\eta > 0$ is the learning rate. The *gain on the target objective* from this update is:

$$\text{Gain}(w) := -g_{\mathcal{T}}^\top \Delta\theta(w) = \eta g_{\mathcal{T}}^\top \mathbf{P} \mathbf{G} w = \eta \sum_{i=1}^{N_C} w_i g_{\mathcal{T}}^\top \mathbf{P} g_i.$$

Different choices of $\mathbf{P}$ recover different data selection and fine-tuning methods, as we now describe.

B.4.1 Special Cases of the Preconditioner

SGD: $\mathbf{P} = \mathbf{I}$. A single stochastic gradient step on example z_i gives:

$$\Delta\theta(e_i) = -\eta g_i,$$

corresponding to $\mathbf{P} = \mathbf{I}$. The gain is:

$$\text{Gain}_{\text{SGD}}(w) = \eta g_{\mathcal{T}}^\top \mathbf{G} w = \eta \sum_i w_i g_{\mathcal{T}}^\top g_i,$$

which is plain Euclidean gradient alignment. This is a first-order approximation to the influence function that ignores training loss curvature entirely.

Influence / Newton: $\mathbf{P} = \mathbf{H}^{-1}$. Let $\mathbf{H} := \nabla_{\theta}^2 \frac{1}{N_C} \sum_{i=1}^{N_C} \ell(z_i; \theta_0)$ denote the training Hessian at the base model. Setting $\mathbf{P} = \mathbf{H}^{-1}$ recovers the influence function exactly:

$$\Delta\theta = -\eta \mathbf{H}^{-1} g_i, \quad \text{Gain}_{\text{inf}}(w) = \eta \sum_i w_i g_{\mathcal{T}}^\top \mathbf{H}^{-1} g_i.$$

This corresponds to re-optimizing to convergence after adding the weighted training points, and uses the training loss curvature $\mathbf{H}$ as the geometry.

Adam: $\mathbf{P} = \mathbf{P}_{\text{Adam}}$. Let $g_t := \nabla_{\theta} \ell(z_t; \theta^t)$ denote the stochastic gradient used at optimization step t . The Adam optimizer maintains running estimates of the first moment $m^t \approx \mathbb{E}[g_t]$ and second moment $v^t \approx \mathbb{E}[g_t \odot g_t]$ of this stochastic gradient. At step t , the coordinate-wise update is [26]:

$$\begin{aligned} \theta^{t+1} - \theta^t &= -\eta_t \frac{m^{t+1}}{\sqrt{v^{t+1} + \epsilon}}, \\ m^{t+1} &= \frac{\beta_1 m^t + (1 - \beta_1) g_t}{1 - \beta_1^t}, \\ v^{t+1} &= \frac{\beta_2 v^t + (1 - \beta_2) (g_t \odot g_t)}{1 - \beta_2^t}. \end{aligned}$$

Here β_1 and β_2 are hyperparameters, and $m^0 = 0$ and $v^0 = 0$.

To connect Adam with our preconditioned-gradient framework, we freeze the coordinate-wise scaling at the base model θ_0 . For a candidate example i , recall that $g := \nabla_{\theta} \ell(z_i; \theta_0)$. The effective per-coordinate scaling applied to this raw gradient is

$$(\Delta\theta)_j = -\eta \frac{(g_i)_j}{\sqrt{\mathbb{E}[(g_i)_j^2] + \epsilon}}.$$

Thus Adam induces the diagonal preconditioner

$$\mathbf{P}_{\text{Adam}} = \text{diag} \left(\frac{1}{\sqrt{\mathbb{E}[(g_i)_1^2] + \epsilon}}, \dots, \frac{1}{\sqrt{\mathbb{E}[(g_i)_d^2] + \epsilon}} \right) \in \mathbb{R}^{d \times d}.$$

Geometrically, $\mathbf{P}_{\text{Adam}}$ is a *diagonal approximation* to the inverse Hessian $\mathbf{H}^{-1}$: it rescales each coordinate by the inverse square root of the gradient’s second moment, using elementwise variance as a proxy for curvature. Directions with high gradient variance are shrunk, and directions with low gradient variance are amplified. Compared to $\mathbf{H}^{-1}$, which captures all off-diagonal curvature interactions, $\mathbf{P}_{\text{Adam}}$ ignores cross-coordinate dependencies but is computationally tractable at the scale of large language models.

The gain under Adam preconditioning is:

$$\begin{aligned}\text{Gain}_{\text{Adam}}(w) &= \eta g_{\mathcal{T}}^{\top} \mathbf{P}_{\text{Adam}} \hat{\mathbf{G}}^{\text{raw}} w \\ &= \eta \sum_{i=1}^{N_C} w_i g_{\mathcal{T}}^{\top} \mathbf{P}_{\text{Adam}} g_i \\ &= \eta \sum_{i=1}^{N_C} w_i \sum_{j=1}^d \frac{(g_{\mathcal{T}})_j (g_i)_j}{\sqrt{\mathbb{E}[(g_i)_j^2] + \epsilon}}.\end{aligned}$$

This is the coordinate-wise inner product between the target gradient and each training gradient, normalized by the Adam scaling factor.

In practice, $\mathbb{E}[(g_i)_j^2]$ is estimated from a small reference set during a warm-up phase similar to LESS [6]. We can aggregate the second moment over the T steps during this warm-up phase:

$$\mathbf{P}_{\text{Adam}} = \text{diag}\left(\frac{1}{\sqrt{\bar{v}_j + \epsilon}}\right), \quad \bar{v}_j = \frac{1}{T} \sum_{t=1}^T v_j^t,$$

making the score efficient to compute.

Natural gradient: $\mathbf{P} = \mathbf{F}^{-1}$. Replacing $\mathbf{H}$ with the Fisher information matrix $\mathbf{F} = \mathbb{E}_{y \sim p_{\theta_0}(\cdot|x)}[\nabla_{\theta} \log p_{\theta_0} \nabla_{\theta} \log p_{\theta_0}^{\top}]$ gives the natural gradient update $\Delta\theta = -\eta \mathbf{F}^{-1} g_i$, where unit steps are measured by KL divergence rather than Euclidean distance in parameter space.

C Further details on SAFE-DRIFTdata selector in Section 2.3

C.1 Low-Rank Working Basis

All downstream computation is carried out in a K -dimensional working basis that combines the retained reference Fisher eigenspace with a residualized target-side subspace.

Estimating $g_{\mathcal{T}}$ Given the target sample $\mathcal{D}_{\text{tar}} = \{z_j\}_{j=1}^{N_T}$, we estimate the target gradient by the sample mean

$$\hat{g}_{\text{tar}}^{\text{raw}} := \frac{1}{N_T} \sum_{j=1}^{N_T} \nabla_{\theta} \ell(z_j; \theta_0) \in \mathbb{R}^d. \quad (21)$$

This is the direct plug-in estimator of the oracle target gradient.

Reference basis The rank- K_R reference basis $\hat{\mathbf{U}}_{K_R} \in \mathbb{R}^{d \times K_R}$ is estimated as the top- K_R eigenvectors from the dual Gram matrix $\frac{1}{N_R} \mathbf{S}_R \mathbf{S}_R^{\top}$ and satisfies $\hat{\mathbf{U}}_{K_R}^{\top} \hat{\mathbf{U}}_{K_R} = \mathbf{I}_{K_R}$ and $\hat{\mathbf{F}}_R^{(K_R)} \hat{\mathbf{U}}_{K_R} = \hat{\mathbf{U}}_{K_R} \hat{\mathbf{\Lambda}}_{K_R}$.

Residualized task basis Collect the per-example gradients $\hat{g}_i^{\text{raw}}$ for all $i \in [N_T + N_C]$ into

$$\hat{\mathbf{G}}_{TC} := [\hat{g}_1^{\text{raw}} \quad \cdots \quad \hat{g}_{N_C+N_T}^{\text{raw}}] \in \mathbb{R}^{d \times (N_C+N_T)}.$$

A low-rank approximation of the reference Fisher alone is not sufficient to represent target-side structure outside $\text{span}(\hat{\mathbf{U}}_{K_R})$. We therefore build an additional task block by first residualizing against the reference subspace:

$$\tilde{\mathbf{G}}_{TC} := (\mathbf{I}_d - \hat{\mathbf{U}}_{K_R} \hat{\mathbf{U}}_{K_R}^{\top}) \hat{\mathbf{G}}_{TC}.$$

On the residualized matrix, we form the dual Gram matrix $\frac{1}{N_T + N_C} \tilde{\mathbf{G}}_{TC}^\top \tilde{\mathbf{G}}_{TC}$ and compute its top- K_T eigenpairs $(\hat{\mathbf{V}}_{K_T}, \hat{\mathbf{\Omega}}_{K_T})$. We then define the added task basis by

$$\hat{\mathbf{U}}_{K_T} := \tilde{\mathbf{G}}_{TC} \hat{\mathbf{V}}_{K_T} ((N_C + N_T) \hat{\mathbf{\Omega}}_{K_T})^{-1/2} \in \mathbb{R}^{d \times K_T}.$$

By construction, $\hat{\mathbf{U}}_{K_R}^\top \hat{\mathbf{U}}_{K_T} = 0$ and $\hat{\mathbf{U}}_{K_T}^\top \hat{\mathbf{U}}_{K_T} = \mathbf{I}_{K_T}$.

The motivation for this residualization is as follows. If we were to project all quantities onto $\hat{\mathbf{U}}_{K_R}$ alone, the quality of the approximation would depend entirely on how much of the target and candidate gradient signal lies within the reference eigenspace. When the target task and the reference distribution are largely disentangled in parameter space, the target and candidate gradients may be nearly orthogonal to $\hat{\mathbf{U}}_{K_R}$. Projecting onto the reference basis alone would then collapse all gradient coordinates to near zero, making every candidate look equally uninformative. The residual basis $\hat{\mathbf{U}}_{K_T}$ captures the task-relevant directions that survive after the reference subspace is removed, ensuring that the common working basis spans enough of both the reference geometry and the task gradient signal for the selection scores to be meaningful.

Common working basis Stacking the two orthonormal bases gives (9). In this basis, the ridge-regularized metric is,

$$\mathbf{M}_{\alpha, K} := \hat{\mathbf{U}}_K \hat{\mathbf{U}}_K^\top \hat{\mathbf{F}}_R \hat{\mathbf{U}}_K \hat{\mathbf{U}}_K^\top + \alpha \mathbf{I} = \begin{bmatrix} \hat{\mathbf{\Lambda}}_{K_R} + \alpha \mathbf{I}_{K_R} & 0 \\ 0 & \alpha \mathbf{I}_{K_T} \end{bmatrix}. \quad (22)$$

Choosing K_R and K_T The truncation level K_R should be chosen based on how much each eigendirection modifies $(\hat{\mathbf{F}}_R + \alpha \mathbf{I})^{-1}$ relative to the $\alpha^{-1} \mathbf{I}$. Let the nonzero eigenvalues of the empirical reference Fisher $\hat{\mathbf{F}}_R$ be $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \dots \geq \hat{\lambda}_{r_R} > 0$, with corresponding orthonormal eigenvectors $\hat{u}_1, \dots, \hat{u}_{r_R}$ and rank $0 < r_R \leq \min\{N_R, d\}$. Then,

$$(\hat{\mathbf{F}}_R + \alpha \mathbf{I})^{-1} = \alpha^{-1} \mathbf{I} - \sum_{j=1}^{r_R} \frac{\hat{\lambda}_j}{\alpha(\hat{\lambda}_j + \alpha)} \hat{u}_j \hat{u}_j^\top.$$

The j -th direction contributes through the relative correction factor

$$r_j(\alpha) := \frac{\hat{\lambda}_j}{\hat{\lambda}_j + \alpha}. \quad (23)$$

When $\hat{\lambda}_j \ll \alpha$, we have $r_j(\alpha) \approx 0$, so this direction behaves almost isotropically after ridge regularization and may be safely discarded.

Accordingly, in practice, we can choose K_R by

$$K_R^{\text{auto}} := \min \left\{ K : \frac{\sum_{j=1}^K r_j(\alpha)}{\sum_{j=1}^{r_R} r_j(\alpha)} \geq 1 - \delta_R \right\}.$$

Similarly, let the positive eigenvalues of $\frac{1}{N_T + N_C} \tilde{\mathbf{G}}_{TC}^\top \tilde{\mathbf{G}}_{TC}$ be $\hat{\omega}_1 \geq \dots \geq \hat{\omega}_{r_T} > 0$. We then define

$$K_T^{\text{auto}} := \min \left\{ K : \frac{\sum_{j=1}^K \hat{\omega}_j}{\sum_{j=1}^{r_T} \hat{\omega}_j} \geq 1 - \delta_T \right\},$$

for some constant $\delta_T \in (0, 1)$

Choosing α based on (ρ, ϵ) . In practice, we do not have access to those population quantities. We replace $\mathbf{F}_R$ by the low-rank estimate as discussed in Section 2.3, replace g_T by the sample mean target gradient, and work in the reduced basis from Appendix C.1. This yields an empirical counterpart $\hat{\phi}(\alpha)$ with the same monotonicity (see Proposition 3, so $\hat{\alpha}$ can be recovered by bisection on $\hat{\phi}(\alpha) = 2\rho/\epsilon^2$ given by the cost ρ and ϵ). The detailed algorithm is in Algorithm 2.

Algorithm 2 Computing $\hat{\alpha}$ from (ρ, ϵ)

Input: Target feature $\hat{b}_{\text{tar}}^{(K)}$, eigenvalues of reference fisher $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \dots \geq \hat{\lambda}_{K_R}$, drift budget $\rho > 0$, norm radius $\epsilon > 0$, numerical floor $\alpha_{\min} > 0$, initial upper guess $\alpha_{\text{hi}} > 0$, cap $\alpha_{\max} > \alpha_{\min}$, tolerance $\tau > 0$.
Output: Damping parameter $\hat{\alpha}$.

- 1: Compute
$$\hat{c}_j \leftarrow [\hat{b}_{\text{tar}}^{(K)}]_j \quad (j = 1, \dots, K_R).$$
- 2: Define
$$r \leftarrow \frac{2\rho}{\epsilon^2}, \quad \phi(\alpha) \leftarrow \frac{\sum_{j=1}^{K_R} \frac{\hat{\lambda}_j(\hat{c}_j)^2}{(\hat{\lambda}_j + \alpha)^2}}{\sum_{j=1}^{K_R} \frac{(\hat{c}_j)^2}{(\hat{\lambda}_j + \alpha)^2}}, \quad \phi_{\max} \leftarrow \frac{\sum_{j=1}^{K_R} \hat{\lambda}_j(\hat{c}_j)^2}{\sum_{j=1}^{K_R} (\hat{c}_j)^2}.$$
- 3: **if** $r \leq \phi(\alpha_{\min})$ **then**
- 4: **return** $\hat{\alpha} \leftarrow \alpha_{\min}$ $\triangleright$ status $\leftarrow$ too_strict
- 5: **else if** $r \geq \phi_{\max}$ **then**
- 6: **return** $\hat{\alpha} \leftarrow \alpha_{\max}$ $\triangleright$ status $\leftarrow$ inactive
- 7: **else**
- 8: $\ell \leftarrow \alpha_{\min}$
- 9: $h \leftarrow \max(\alpha_{\text{hi}}, 2\alpha_{\min})$
- 10: **while** $\phi(h) < r$ and $h < \alpha_{\max}$ **do**
- 11: $h \leftarrow \min(2h, \alpha_{\max})$
- 12: **end while**
- 13: **for** $t = 1, \dots, T$ **do**
- 14: $\alpha_{\text{mid}} \leftarrow \sqrt{\ell h}$ $\triangleright$ log-space bisection
- 15: $r_{\text{mid}} \leftarrow \phi(\alpha_{\text{mid}})$
- 16: **if** $\frac{|r_{\text{mid}} - r|}{\max(|r|, 10^{-30})} \leq \tau$ **then**
- 17: $\hat{\alpha} \leftarrow \alpha_{\text{mid}}$
- 18: **break**
- 19: **else if** $r_{\text{mid}} < r$ **then**
- 20: $\ell \leftarrow \alpha_{\text{mid}}$
- 21: **else**
- 22: $h \leftarrow \alpha_{\text{mid}}$
- 23: **end if**
- 24: $\hat{\alpha} \leftarrow \alpha_{\text{mid}}$ $\triangleright$ status $\leftarrow$ feasible
- 25: **end for**
- 26: **end if**

C.1.1 Scaling value for $\Delta\theta^*$

For fixed α , we define $\mathbf{M}_{\alpha, K}$ in (22) and $\hat{\mathbf{F}}_R := \text{diag}(\hat{\mathbf{\Lambda}}_{K_R}, \mathbf{0}_{K_T})$. Let the unscaled reduced safe direction be

$$\Delta\theta_K := \mathbf{U}_K \mathbf{M}_{\alpha, K}^{-1} \hat{b}_{\text{tar}}^{(K)}.$$

The practical reduced safe update is then written as

$$\hat{\Delta}\theta_K^* = -\frac{1}{\hat{\mu}_K} \Delta\theta_K = -\frac{1}{\hat{\mu}_K} \mathbf{U}_K \mathbf{M}_{\alpha, K}^{-1} \hat{b}_{\text{tar}}^{(K)}.$$

If a reference-cost budget ρ and/or an ℓ_2 -budget ϵ is imposed, we choose the smallest scaling factor that satisfies all active constraints:

$$\hat{\mu}_K := \max \left\{ 1, \sqrt{\frac{\frac{1}{2} \Delta\theta_K^\top \mathbf{F}_R \Delta\theta_K}{\rho}}, \frac{\|\Delta\theta_K\|_2}{\epsilon} \right\}. \quad (24)$$

Equivalently, the second term is included only when a reference-cost budget ρ is enforced, and the third term is included only when an ℓ_2 -budget ϵ is enforced. Thus, for fixed α , the direction is

determined by $\mathbf{M}_{\alpha,K}^{-1} \hat{b}_{\text{tar}}^{(K)}$, while $\hat{\mu}_K$ provides the minimal global rescaling needed to satisfy the active budgets.

This practical construction should be contrasted with the oracle formulation $\Delta\theta^* = -\epsilon \frac{(\mathbf{F}_{\mathcal{R}} + \alpha \mathbf{I})^{-1} g_{\mathcal{T}}}{\|(\mathbf{F}_{\mathcal{R}} + \alpha \mathbf{I})^{-1} g_{\mathcal{T}}\|_2}$ in which α is itself chosen so that the normalized update satisfies the reference-cost constraint $\frac{1}{2}(\Delta\theta^*)^\top \mathbf{F}_{\mathcal{R}} \Delta\theta^* = \rho$.

C.2 Selection Algorithm

Recall the practical continuous safe update $\hat{\Delta}\theta_K^*(\alpha)$ and the subset-induced reduced update $\hat{\Delta}\theta_K(w)$ from (11) and (12), respectively. Define the whitened candidate-update matrix and target update by

$$\begin{aligned}\tilde{\Delta}_K &:= \mathbf{M}_{\alpha,K}^{1/2} \left(-\frac{\eta}{S} \mathbf{P}_K \mathbf{B}_K \right) \in \mathbb{R}^{K \times N_C}, \\ \tilde{\Delta}\theta_K^* &:= \mathbf{M}_{\alpha,K}^{1/2} \hat{\Delta}\theta_K^*(\alpha).\end{aligned}$$

Then the matching objective

$$\hat{w} \in \arg \min_{\substack{w \in \{0,1\}^{N_C} \\ \mathbf{1}^\top w = S}} \frac{1}{2} \left\| \hat{\Delta}\theta_K(w) - \hat{\Delta}\theta_K^*(\alpha) \right\|_{\mathbf{M}_{\alpha,K}}^2$$

can be rewritten as the whitened objective

$$\hat{w} \in \arg \min_{\substack{w \in \{0,1\}^{N_C} \\ \mathbf{1}^\top w = S}} \frac{1}{2} \left\| \tilde{\Delta}_K w - \tilde{\Delta}\theta_K^* \right\|_2^2.$$

For a singleton subset $w = e_i$, we have

$$\frac{1}{2} \left\| \tilde{\Delta}_K(i) - \tilde{\Delta}\theta_K^* \right\|_2^2 = \frac{1}{2} \left\| \tilde{\Delta}\theta_K^* \right\|_2^2 - \left\langle \tilde{\Delta}_K(i), \tilde{\Delta}\theta_K^* \right\rangle + \frac{1}{2} \left\| \tilde{\Delta}_K(i) \right\|_2^2.$$

Since the first term is independent of i , ranking singleton subsets by smaller objective value is equivalent to ranking them by

$$s_i := \left\langle \tilde{\Delta}_K(i), \tilde{\Delta}\theta_K^* \right\rangle - \frac{1}{2} \left\| \tilde{\Delta}_K(i) \right\|_2^2.$$

Thus, s_i is exactly the decrease in the whitened quadratic objective obtained by selecting candidate i alone. Compared with LESS [6], which ranks examples by alignment alone, our selector adds the quadratic cost term to keep the update in a reference-stable geometry. We provide the complete selection algorithm in Algorithm 3.

D Experiments

D.1 Synthetic linear experiments

To build better intuition on the performance of our method in simple settings, we begin with a family of synthetic linear models designed to isolate the two statistical ingredients highlighted by our theory: the estimation problem (estimating the target gradient $g_{\mathcal{T}}$ from few target examples, and estimating the reference geometry $\mathbf{F}_{\mathcal{R}}$ from limited reference data) and the optimization problem (finding the best gradient update under off-target budget).

Linear-Gaussian setup. We first consider a simple linear Gaussian model. For an input $x \in \mathbb{R}^d$, the base model is

$$p_{\theta}(y | x) = \mathcal{N}(x^\top \theta, \sigma^2), \quad \ell((x, y); \theta) = \frac{1}{2\sigma^2} (y - x^\top \theta)^2.$$

Target examples are generated by

$$x \sim \mathcal{N}(0, \Sigma_{\mathcal{T}}), \quad y = x^\top \beta_{\mathcal{T}} + \xi, \quad \xi \sim \mathcal{N}(0, \sigma^2),$$

Algorithm 3 Forward greedy solver for the whitened reduced objective

Input: Whitened candidate-update matrix $\tilde{\Delta}_K = [\tilde{\Delta}_K(1) \cdots \tilde{\Delta}_K(N_C)]$, whitened target update $\tilde{\Delta}\theta_K^*$, subset budget S , shortlist size $m \geq S$.

Output: Selected subset $\hat{\mathcal{S}} \subseteq [N_C]$ with $|\hat{\mathcal{S}}| = S$, and selection vector $\hat{w}$.

1: For each $i \in [N_C]$, compute the singleton score

$$s_i \leftarrow \left\langle \tilde{\Delta}_K(i), \tilde{\Delta}\theta_K^* \right\rangle - \frac{1}{2} \|\tilde{\Delta}_K(i)\|_2^2.$$

2: Let $\mathcal{M}_m \subseteq [N_C]$ be the indices of the top- m scores $\{s_i\}_{i=1}^{N_C}$.

3: Initialize $\hat{\mathcal{S}} \leftarrow \emptyset$ and $\tilde{s} \leftarrow 0 \in \mathbb{R}^K$.

4: **for** $t = 1, \dots, S$ **do**

5: Choose

$$j^* \leftarrow \arg \min_{j \in \mathcal{M}_m \setminus \hat{\mathcal{S}}} \left(\frac{1}{2} \|\tilde{\Delta}_K(j)\|_2^2 + \left\langle \tilde{s}, \tilde{\Delta}_K(j) \right\rangle - \left\langle \tilde{\Delta}\theta_K^*, \tilde{\Delta}_K(j) \right\rangle \right).$$

6: Update

$$\hat{\mathcal{S}} \leftarrow \hat{\mathcal{S}} \cup \{j^*\}, \quad \tilde{s} \leftarrow \tilde{s} + \tilde{\Delta}_K(j^*).$$

7: **end for**

8: Set $\hat{w}_i \leftarrow \mathbf{1}\{i \in \hat{\mathcal{S}}\}$ for all $i \in [N_C]$.

9: **return** $\hat{\mathcal{S}}, \hat{w}$.

while reference inputs are drawn from

$$x \sim \mathcal{N}(0, \Sigma_{\mathcal{R}}).$$

We set the base point to $\theta_0 = 0$. In this model, the per-example gradient and target gradient at the base point are

$$g_i = \nabla_{\theta} \ell(z_i; \theta_0) = -\frac{1}{\sigma^2} y_i x_i, \quad g_{\mathcal{T}} = \nabla_{\theta} L_{\mathcal{T}}(\theta_0) = -\frac{1}{\sigma^2} \Sigma_{\mathcal{T}} \beta_{\mathcal{T}},$$

and the reference Fisher is

$$\mathbf{F}_{\mathcal{R}} = \mathbb{E}_{x \sim \mathcal{R}} \mathbb{E}_{y \sim p_{\theta_0}(\cdot | x)} [\nabla_{\theta} \log p_{\theta_0}(y | x) \nabla_{\theta} \log p_{\theta_0}(y | x)^{\top}] = \frac{1}{\sigma^2} \Sigma_{\mathcal{R}}.$$

In this synthetic setting, the raw per-example gradient g_i is available in closed form; it plays the same role as $\hat{g}_i^{\text{raw}}$ in the empirical algorithm. Moreover, unlike the general nonlinear setting where the quadratic expansion is local, the reference drift is *exact* here:

$$D_{\mathcal{R}}(\Delta\theta) = \mathbb{E}_{x \sim \mathcal{R}} [D_{\text{KL}}(p_{\theta_0}(\cdot | x) \| p_{\theta_0 + \Delta\theta}(\cdot | x))] = \frac{1}{2\sigma^2} \Delta\theta^{\top} \Sigma_{\mathcal{R}} \Delta\theta = \frac{1}{2} \Delta\theta^{\top} \mathbf{F}_{\mathcal{R}} \Delta\theta.$$

This makes the synthetic setting especially useful for separating optimization and selection effects from approximation error in the drift model.

Updates and evaluation. Given a selected subset $\mathcal{S} \subset \mathcal{D}_{\text{train}}$ of size S , we form the update

$$\Delta\theta(\mathcal{S}) = -\frac{\eta}{S} \mathbf{P} \sum_{i \in \mathcal{S}} g_i,$$

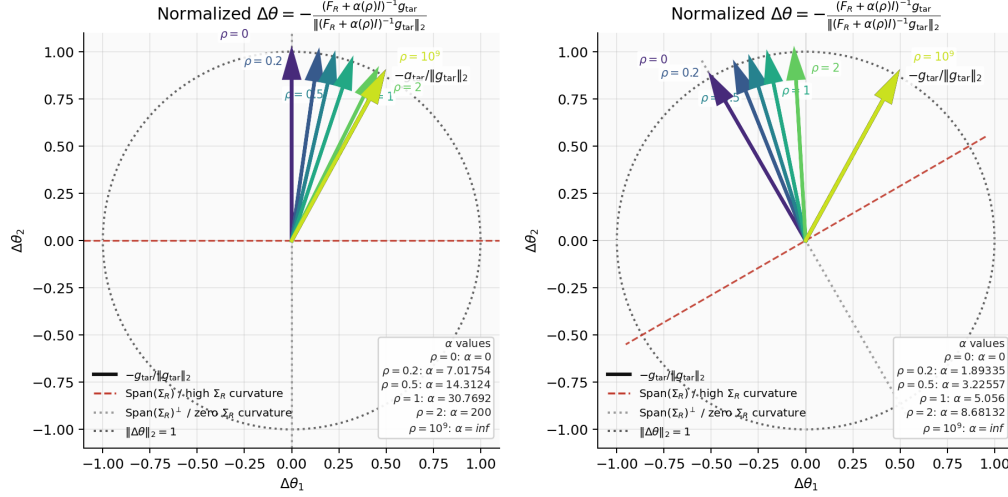
where $\mathbf{P}$ is the preconditioner associated with the update rule. In particular, we use $\mathbf{P} = \mathbf{I}_d$ for stochastic gradient descent. We evaluate each selected update using

$$\text{Gain}_{\text{held}}(\mathcal{S}) = L_{\mathcal{T}}(\theta_0) - L_{\mathcal{T}}(\theta_0 + \Delta\theta(\mathcal{S})),$$

$$\text{Drift}_{\text{held}}(\mathcal{S}) = D_{\mathcal{R}}(\Delta\theta(\mathcal{S})),$$

and, when the population quantities are known, the directional error relative to the oracle update

$$\Delta\theta^* \propto -(\mathbf{F}_{\mathcal{R}} + \alpha \mathbf{I}_d)^{-1} g_{\mathcal{T}}, \quad \text{DirErr}(\mathcal{S}) = 1 - \cos(\Delta\theta(\mathcal{S}), \Delta\theta^*).$$



(a) Results for $\Sigma_{\mathcal{R}}$ parametrized by $(u \ v) = \mathbf{I}_2$. (b) Results for $\Sigma_{\mathcal{R}}$ parametrized by $(u \ v) = \begin{pmatrix} \cos(\pi/6) & -\sin(\pi/6) \\ \sin(\pi/6) & \cos(\pi/6) \end{pmatrix}$.

Figure 4: Visualization of the optimal gradient updates $\Delta\theta^*$, here varying with the damping parameter α . In this experiment, $\Sigma_{\mathcal{T}} = \mathbf{I}_2$, while $\Sigma_{\mathcal{R}} = 20uu^\top + 0.5vv^\top$, for different choices of u, v . The “forbidden” direction is highlighted in red. The target distribution $\Sigma_{\mathcal{T}}$ and the forbidden space $\Sigma_{\mathcal{R}}$ have the same span, but $\Sigma_{\mathcal{R}}$ puts a high cost on projections along the red axis, inducing a shift in the optimal gradient under budget constraints.

Experiment 1: Low-dimensional geometry ($d = 2$). We first consider a two-dimensional setting in which the geometry can be visualized directly. We choose

$$\Sigma_{\mathcal{R}} = \mathbf{U} \text{diag}(25, 1) \mathbf{U}^\top, \quad \Sigma_{\mathcal{T}} = \mathbf{I}_2, \quad \beta_{\mathcal{T}} = (1, 1)^\top, \quad \sigma^2 = 1, \quad (25)$$

where $\mathbf{U}$ is a rotation matrix with angle $\pi/6$. In this setting, one can plot in the $(\Delta\theta_1, \Delta\theta_2)$ plane: (i) level sets of the target gain $-g_{\mathcal{T}}^\top \Delta\theta$, (ii) ellipses of constant reference drift $D_{\mathcal{R}}(\Delta\theta)$, and (iii) the induced update directions for different methods and damping values α . The results are provided in Figure 4. This experiment provides a direct geometric picture of how the reference Fisher $\mathbf{F}_{\mathcal{R}}$ rotates and rescales the target direction, inducing certain directions (here, the direction parallel to the axis in red) to have high cost.

We first consider a synthetic linear model where target samples follow $x \sim \mathcal{N}(0, \Sigma_{\mathcal{T}})$, $y = x^\top \beta_{\mathcal{T}} + \xi$ with coefficients $\beta_{\mathcal{T}}$ and noise $\xi \sim \mathcal{N}(0, \sigma^2)$ and reference inputs $x \sim \mathcal{N}(0, \Sigma_{\mathcal{R}})$, to gain an intuition of SAFE-DRIFT performance under two settings: an *entangled setting* where $-g_{\mathcal{T}}$ projects strongly on $\mathbf{F}_{\mathcal{R}}$, and a *disentangled setting* where its projection along $\mathbf{F}_{\mathcal{R}}$ is weak. Additional details on the data-generating procedure are provided in Appendix D.1. From Figure 5, we see that SAFE-DRIFT has to navigate a real tradeoff between target gradient alignment and off-target budget because of the entanglement of $\Sigma_{\mathcal{R}}$ and $-g_{\mathcal{T}}$. By contrast, in the disentangled regime where the update direction $-g_{\mathcal{T}}$ lies primarily in a low-cost direction, SAFE-DRIFT stays closer to the LESS baseline that aims to maximize the target gain.

Experiment 2: Finite-pool subset selection in $d = 2$. To visualize which points each method would preferentially select, we consider the setting of Experiment 1 (Equation 25) and we generate a finite candidate pool from a three-component Gaussian mixture:

$$x_i \sim \frac{1}{3}\mathcal{N}(0, \Sigma_1) + \frac{1}{3}\mathcal{N}(0, \Sigma_2) + \frac{1}{3}\mathcal{N}(0, \Sigma_3), \quad y_i = x_i^\top \beta_{\mathcal{T}} + \xi_i, \quad \xi_i \sim \mathcal{N}(0, \sigma^2), \quad (26)$$

with

$$\Sigma_1 = \mathbf{U} \text{diag}(9, 0.25) \mathbf{U}^\top, \quad \Sigma_2 = \mathbf{U} \text{diag}(0.25, 9) \mathbf{U}^\top, \quad \Sigma_3 = \mathbf{I}_2.$$

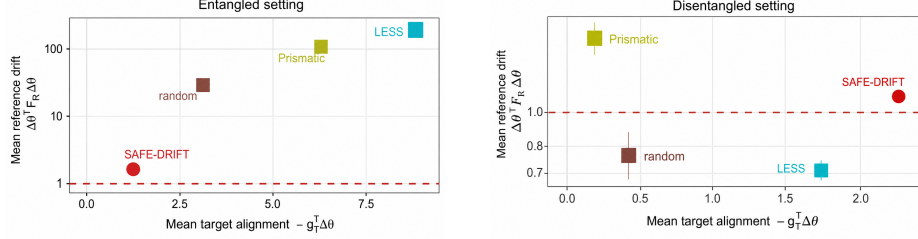


Figure 5: Trade-off between gradient alignment and budget constraints for an “entangled regime” and a “disentangled regime”. The results are averaged over 100 runs. The red line denotes the budget ρ . SAFE-DRIFT is the only method able to control the drift.

The first component produces candidate gradients aligned with the high-curvature directions of $\mathbf{F}_{\mathcal{R}}$, the second produces gradients aligned with low-curvature directions, and the third supplies isotropic directions. The raw data — and the corresponding gradients of each point — are shown in Figure 6. For each method, we visualize the pool of candidate gradients, highlight the selected subset, and plot the resulting update vector (Figure 7).

Experiment 3: Tradeoffs in Alignment/ off-target drift in $d = 2$. We visualize the tradeoff between target alignment and reference drift using the previous experimental setting. Here, we set

$$\beta_{\mathcal{T}} = (-\sin(\pi/6), \cos(\pi/6)) = \left(-\frac{1}{2}, \frac{\sqrt{3}}{2}\right),$$

For each selected subset we report both exact and held-out metrics. The exact first-order alignment is

$$-g_{\mathcal{T}}^{\top} \Delta \theta,$$

the exact target improvement is

$$\text{TI}(\Delta \theta) = -g_{\mathcal{T}}^{\top} \Delta \theta - \frac{1}{2} \Delta \theta^{\top} \mathbf{F}_{\mathcal{R}} \Delta \theta,$$

and the exact reference drift is

$$\text{Drift}(\Delta \theta) = \frac{1}{2} \Delta \theta^{\top} \mathbf{F}_{\mathcal{R}} \Delta \theta.$$

In addition, we evaluate held-out target improvement and held-out reference drift on fresh synthetic samples to measure out-of-sample behavior. The main aggregate plots in the appendix average these quantities over 100 independent random seeds and report standard errors. The default “entangled” setting uses $\beta_{\mathcal{T}} = (1, 1)$, while the “disentangled” or orthogonal variant uses

$$\beta_{\mathcal{T}} = (-\sin(\pi/6), \cos(\pi/6)) = \left(-\frac{1}{2}, \frac{\sqrt{3}}{2}\right),$$

which is nearly orthogonal to the dominant reference direction. This second variant isolates the regime in which target-relevant directions and reference-sensitive directions are deliberately misaligned.

We report both gradient alignment and held-out reference drift, which together reveal the tradeoff between aggressive target alignment and robustness to off-target effects. In this case, because of the entanglement of $\Sigma_{\mathcal{R}}$ and $-g_{\mathcal{T}}$, SAFE-DRIFT has to navigate a real tradeoff between gradient alignment and budget constraints. By contrast, in the disentangled regime where the update direction $-g_{\mathcal{T}}$ lies primarily in a low-cost direction, SAFE-DRIFT stays closer to the LESS baseline.

Effect of estimating $g_{\mathcal{T}}$. We evaluate subset selection in a synthetic high-dimensional teacher-student model where the reference and target geometries can have different spectral structure. In all runs, the ambient dimension is $d = 200$, the true reference rank is $r = 40$, the candidate pool contains $N_C = 800$ examples, the target gradient is estimated from $N_T = 32$ samples, the reference Fisher is estimated from $N_R = 300$ samples, and the subset budget is $S = 40$. We generate the reference Fisher as $\mathbf{F}_{\mathcal{R}} = \sigma^{-2} \mathbf{U}_R \text{diag}(\lambda^R) \mathbf{U}_R^{\top}$ and the target covariance as $\Sigma_{\mathcal{T}} = \mathbf{U}_T \text{diag}(\lambda^T) \mathbf{U}_T^{\top}$, where $\mathbf{U}_R$ and $\mathbf{U}_T$ are independent random orthonormal bases. We sweep both the reference and target spectra over geometric, linear, and power-law decay, and in a denser

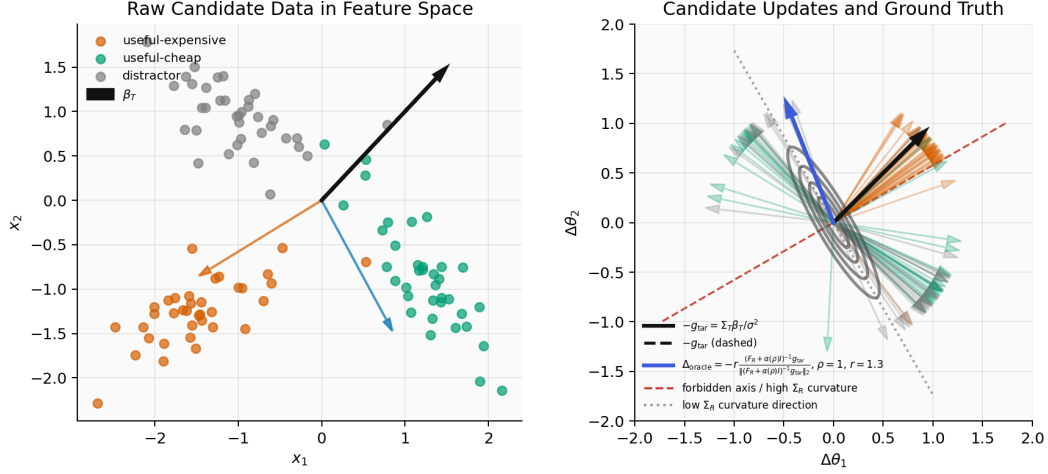


Figure 6: Visualization of the raw random design (X) distribution (left) in the setting of Experiment 2 (Eq. 26) and the corresponding gradient (normalized to visualize direction, right). Note that in this setting $-g_T = \frac{1}{\sigma^2} \Sigma_T \beta_T = \beta_T$, so the black arrow indicates the direction of the update $-g_T$.

variant we sweep power-law exponents $(q_R, q_T) \in \{0.5, 0.75, 1.0, 1.5, 2.0\}^2$. For each spectrum pair, we compare SAFE-DRIFTfull, SAFE-DRIFTlow-rank, SAFE-DRIFTreference-only, SAFE-DRIFTtask-only, SAFE-DRIFTrandom- K , SAFE-DRIFTdiagonal, LESS, random, and Prismatic over requested working ranks $K_R + K_T \in \{10, 20, 40, 60, 80, 120\}$. All SAFE-DRIFTmethods are run under a fixed reference-drift budget $\rho = 1$, and the realized subset update is always the SGD-style average $\Delta\theta(\mathcal{S}) = -(\eta/S) \sum_{i \in \mathcal{S}} g_i$. We report the exact realized reference drift $\frac{1}{2} \Delta\theta^\top \mathbf{F}_R \Delta\theta$, the exact target gain, and the absolute regret relative to SAFE-DRIFTfull, averaged over 10 random seeds and 3 target regimes (for a total of 30 instances per spectrum pair), with standard-error bars shown in all plots.

D.2 Entanglement diagnostics in the working subspace

D.2.1 Entanglement quadrant

A key insight from Proposition 2 is that entanglement between the target gradient and the reference Fisher geometry determines when drift control is most critical to performance on the target task. To turn this into practical insights, we use the low-rank basis $\mathbf{U}_K = [\hat{\mathbf{U}}_{K_R} \ \hat{\mathbf{U}}_{K_T}]$ to measure whether a selected candidate improves the target task in a way that is entangled with off-target behavior. Decompose the candidate- and target-gradient coordinates in this basis as,

$$\hat{b}_i^{(K)} = \mathbf{U}_K^\top \hat{g}_i^{\text{raw}} = \begin{bmatrix} \hat{\mathbf{U}}_{K_R}^\top \hat{g}_i^{\text{raw}} \\ \hat{\mathbf{U}}_{K_T}^\top \hat{g}_i^{\text{raw}} \end{bmatrix} = \begin{bmatrix} \hat{b}_i^R \\ \hat{b}_i^T \end{bmatrix}, \quad \hat{b}_{\text{tar}}^{(K)} = \mathbf{U}_K^\top \hat{g}_{\text{tar}}^{\text{raw}} = \begin{bmatrix} \mathbf{U}_{K_R}^\top \hat{g}_{\text{tar}}^{\text{raw}} \\ \mathbf{U}_{K_T}^\top \hat{g}_{\text{tar}}^{\text{raw}} \end{bmatrix} = \begin{bmatrix} \hat{b}_{\text{tar}}^R \\ \hat{b}_{\text{tar}}^T \end{bmatrix}.$$

Here, $\hat{b}_i^R$ refers to the projection of the candidate gradients along the reference space, while $\hat{b}_i^T$ refers to their projection in the residualized target space. We define two target-alignment scores under the reference subspace and the residual task subspace, respectively,

$$d_i^R = (\hat{b}_i^R)^\top (\hat{\Lambda}_{K_R} + \alpha \mathbf{I}_{K_R})^{-1} \hat{b}_{\text{tar}}^R, \quad d_i^T = \alpha^{-1} (\hat{b}_i^T)^\top \hat{b}_{\text{tar}}^T.$$

Here, $d_i^R > 0$ indicates that the candidate gradient aligns target signal in the reference geometry, while $d_i^T > 0$ indicates that the gradient projects on the target signal in directions decoupled from the reference-sensitive directions. We can thus partition candidates by the signs of (d_i^R, d_i^T) :

$$Q_{\text{ref}} := \{z_i \in \mathcal{D}_{\text{train}} : d_i^T \leq 0, d_i^R > 0\}, \quad Q_{\text{clean}} := \{z_i \in \mathcal{D}_{\text{train}} : d_i^T > 0, d_i^R \leq 0\},$$

$$Q_{\text{ent}} := \{z_i \in \mathcal{D}_{\text{train}} : d_i^T > 0, d_i^R > 0\}, \quad Q_{\text{mis}} := \{z_i \in \mathcal{D}_{\text{train}} : d_i^T \leq 0, d_i^R \leq 0\}.$$

We then measure the proportions of the candidate examples in each quadrant by,

$$\pi_Q := \frac{1}{|N_C|} \sum_{z_i \in \mathcal{D}_{\text{train}}} \mathbf{1}\{z_i \in Q\}, \quad Q \in \{Q_{\text{clean}}, Q_{\text{ent}}, Q_{\text{ref}}, Q_{\text{mis}}\}. \quad (27)$$

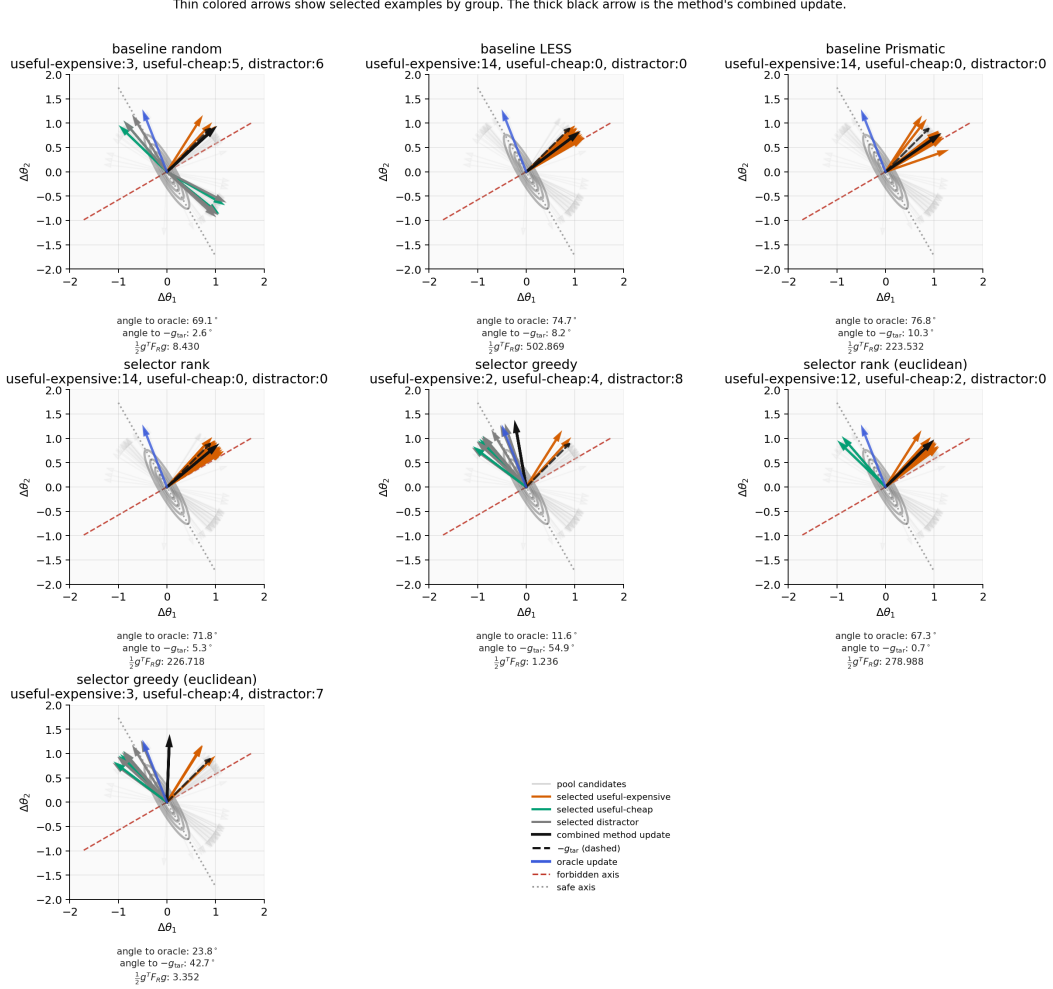


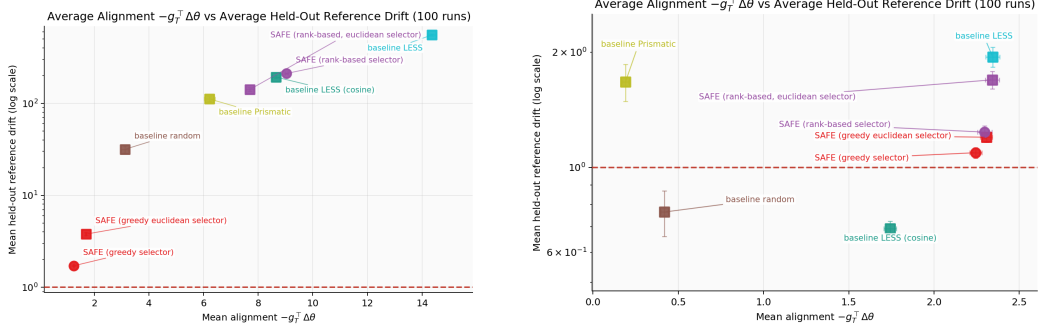
Figure 7: Comparison of data selection methods

These four proportions summarize the entanglement structure of the candidate pool. Specifically, a large $\pi_{Q_{\text{ent}}}$ implies a positive target alignment is strongly tied to reference-sensitive directions, so target improvement is more likely to come at the cost of off-target drift. A large $\pi_{Q_{\text{clean}}}$ indicates that many candidates carry target signal through residual task directions that are less coupled to the reference geometry. In contrast, Q_{ref} captures candidates whose target alignment exists only in the reference-entangled component, and Q_{mis} captures candidates with non-positive alignment in both components. We provide $\pi_{Q_{\text{clean}}}$ and $\pi_{Q_{\text{ent}}}$ of our experiments in Table 1.

Importantly, the quadrant percentages in (27) are descriptive diagnostics, not selection rules. SAFE-DRIFT does not simply select all examples from Q_{clean} : some examples in Q_{ent} may carry necessary target signal when the target gradient is genuinely entangled with the reference geometry, and some examples in Q_{clean} may be redundant or negligibly small in magnitude. The quadrant percentages, therefore, explain how the selected subset uses residual versus reference-entangled target signals.

D.2.2 Reference-entanglement diagnostics

Moreover, the decomposition in $\hat{b}_i^{(K)}$ to $\hat{b}_i^R$ and $\hat{b}_i^T$ makes entanglement analysis possible in reduced coordinates. The entanglement analysis asks whether the *target-aligned contribution* is carried primarily through the reference-sensitive block $\hat{b}_i^R$ or through the residual task block $\hat{b}_i^T$. The former is more likely to consume reference budget, whereas the latter reflects target improvement with less direct pressure on reference behavior.



(a) Entangled regime: $-g_T$ projects strongly on the “forbidden” direction (U_1). (b) Disentangled regime: $-g_T \propto U_2$ projects weakly on the “forbidden” direction.

Figure 8: Tradeoff between gradient alignment and budget constraints. Results are averaged over 100 independent experiments. Uncertainty is denoted by error bars, which are on occasions smaller than the size of the markers. The red line at $y = 1$ denotes the budget. SAFE-DRIFT is the only method able to control the drift.

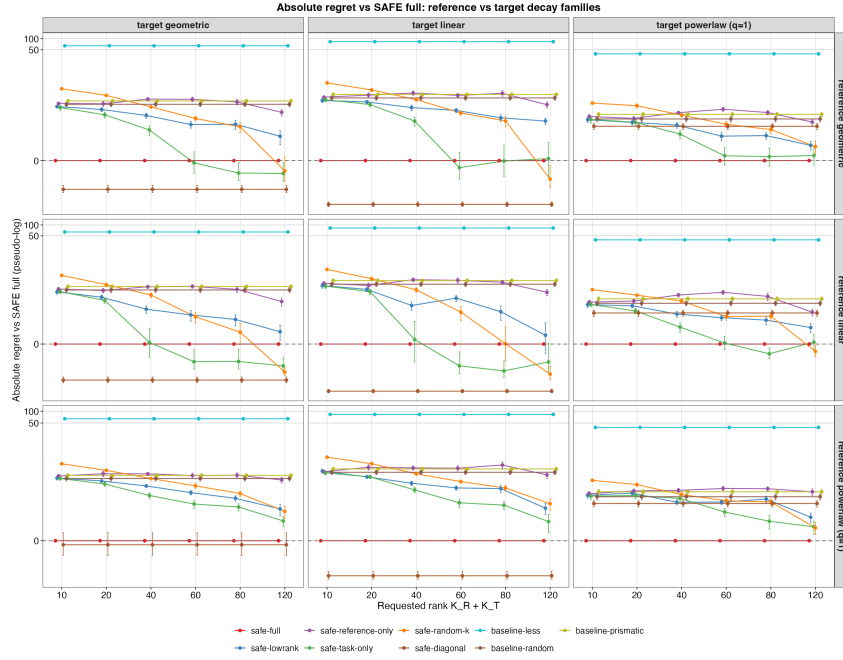


Figure 9: Regret for the baselines and different variants of SAFE-DRIFT (colors), per requested rank (x -axis) as a function of the spectrum of g_T and F_R .

We use two complementary *reference-entanglement diagnostics* for a reduced gradient

$$\rho_E(b) = \frac{\|b^R\|_2^2 / K_R}{\|b^R\|_2^2 / K_R + \|b^T\|_2^2 / K_T}, \quad \rho_{\text{safe}}(b) = \frac{(b^R)^\top \Lambda_R b^R / K_R}{(b^R)^\top \Lambda_R b^R / K_R + \alpha \|b^T\|_2^2 / K_T}. \quad (28)$$

The quantity $\rho_E(b)$ is a purely geometric measure of how much of the gradient lies in the reference block versus the residual task block, while $\rho_{\text{safe}}(b)$ measures the same allocation under the safe-update geometry. In this sense, both quantities measure how strongly the gradient is *entangled* with the reference-sensitive subspace.

We apply these diagnostics to both the target gradient and the candidate gradients additional to the quantities in Section D.2.1. For the target gradient, a large reference share indicates that the target objective itself is entangled with reference-sensitive directions, making aggressive target alignment inherently drift-prone. For candidate gradients, a large reference share indicates that the

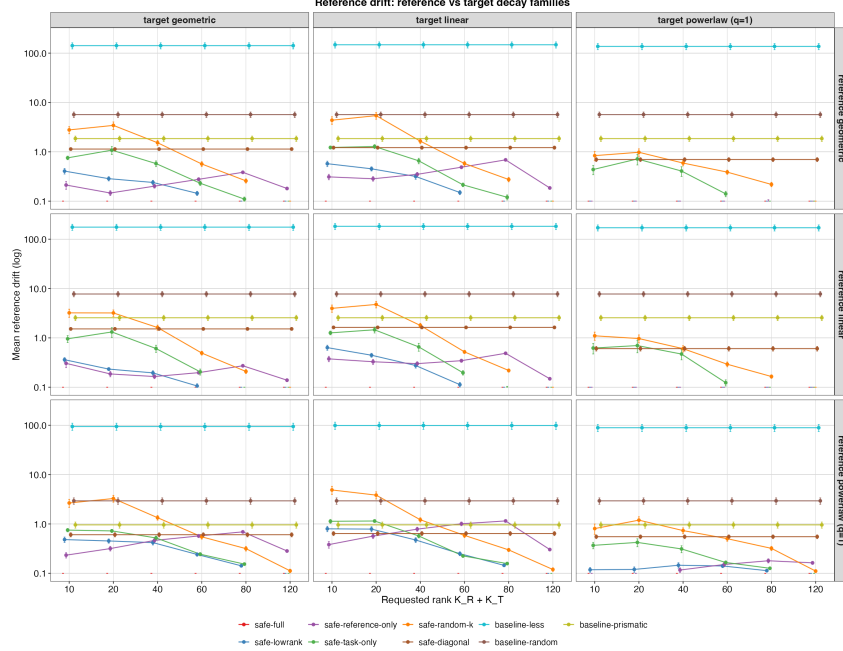


Figure 10: Off-target drift for the baselines and different variants of SAFE-DRIFT(colors), per requested rank (x-axis) as a function of the spectrum of $g_{\mathcal{T}}$ and $\mathbf{F}_{\mathcal{R}}$.

example pushes the model strongly along reference-sensitive directions and is therefore more likely to contribute to off-target drift. Importantly, these diagnostics are most informative when interpreted together with target alignment in Section D.2.1. These

Because these quantities are computed in the same structured working subspace used by our selection algorithm, they are inexpensive to evaluate and directly tied to the update geometry of selection. This is a key advantage of the structured low-rank basis. Compared to a generic random sketch, our low rank projection separates reference-sensitive and residual task-relevant directions, making blockwise entanglement analysis both stable and interpretable.

D.3 Additional details on experiments

All experiments share the following configuration. The candidate pool $\mathcal{D}_{\text{train}}$ is split into a 3% warmup subset used to estimate the Adam preconditioner $\mathbf{P}$ and a remaining 97% from which training points are selected. The target set $\mathcal{D}_{\text{tar}}$ is divided into three thirds: one for computing the target gradient $\hat{g}_T$, one for validation, and one as a held-out test set. The reference set $\mathcal{D}_{\text{ref}}$ is split evenly between estimating $\hat{\mathbf{F}}_R$ and evaluating reference drift. Each selected subset is then used for fine-tuning with the LoRA training recipe in Table 1 using dropout 0.1, target modules $\{q, k, v, o\}$, Adam update with learning rate 2×10^{-4} , 1 epoch, effective batch size 32, and sequence length 4096. Gradients for selection and model updates are computed over LoRA adapters [30] applied to the attention modules.

Compute Resources For model fine-tuning, we use Huggingface Accelerate [64] and utilize Zero3 optimizer in DeepSpeed on 2 A100 GPUs.

D.3.1 Baseline selectors

We compare SAFE-DRIFT against five baselines. Full finetunes on the entire candidate pool without selection, and Random uniformly subsamples the pool at the target budget.

All gradient-based methods—LESS, Prismatic Synthesis, and SAFE-DRIFT—operate on the same set of low-rank projected gradients computed in the shared working basis $\mathbf{U}_K$, ensuring that differences in selection performance reflect the selection criterion rather than gradient featurization. LESS [6]

scores each candidate by the cosine similarity between its projected gradient and the projected target gradient, using the low-rank Adam-style preconditioner as the inner product metric. We adapt Prismatic Synthesis [19] to promote diversity by clustering candidate gradients via k -means and sampling inversely proportional to cluster size, with the number of clusters set to 10% of the pool and 20 Lloyd iterations.

DSIR [17] operates on hashed n -gram features rather than gradients, scoring candidates by the dot product between their hashed n -gram count vectors and the log importance ratio $\log p - \log q$, estimated from a 30% subsample of the pool with 10,000 hash buckets.

D.3.2 Setting 1: Code generation

The goal is to improve Python programming ability while leaving the base model’s behavior on stereotypes unchanged. The candidate pool is CodeAlpaca [34], a 20K dataset of general-purpose programming instructions generated via self-instruct [65]. The target set is HumanEval [35], a benchmark of 164 hand-written Python programming problems; we split these into 109 test, 27 validation, and 28 examples for computing the target gradient $\hat{g}_T$. The reference distribution is defined by the intrasentence split of StereoSet [36], which presents each prompt with a stereotypical, an anti-stereotypical, and an unrelated completion. We retain the stereotypical and anti-stereotypical completions in equal proportion to construct a balanced reference set. This results in 4.21K samples which we split into half for reference Fisher estimation and for evaluating off-target drift. Since computing gradients over the full candidate pool is memory-intensive, the task-side basis $\hat{\mathbf{U}}_{K_T}$ is estimated from 4,000 randomly subsampled candidates. We ablate over three low-rank settings: $(K_R, K_T) \in \{(128, 36), (256, 64), (512, 128)\}$ with selection budget 15%; results are reported in Figure 17 and 18.

Evaluation metrics. For the target task, we evaluate code generation quality via $\text{pass}@k$ rates following the protocol of [35]: for each problem, k candidate solutions are sampled and the problem is considered solved if at least one passes all unit tests. We report $\text{pass}@1$ and $\text{pass}@10$ (mean and standard deviation) for all selectors in Table 2. For the reference distribution, we use StereoSet score, defined as the proportion of prompts for which the model assigns higher average token probability to the stereotypical completion than to the anti-stereotypical one [36]. A perfectly unbiased model scores 50; the base model scores 64.9, indicating slight preference for stereotypical associations. We additionally record the probability margin, the difference in average token probability between the stereotypical and anti-stereotypical completions. All reference metrics are reported as changes from the base model.

D.3.3 Setting 2: Improving medical knowledge

The goal of this experiment is to improve medical question-answering ability of a model while leaving the base model’s broader non-medical behavior unchanged. The candidate pool is MedMCQA [37], a large collection of multiple-choice medical exam questions. For the target set, we subsample 256 samples from MedQA [38], which consists of medical licensing-style questions. The reference set combines BBQ [39] and TruthfulQA [40], covering social bias and factual reliability respectively, forming a non-medical alignment distribution.

Evaluation metrics. We measure fine-tuning performance on the target task using MedQA multiple-choice accuracy, reporting development accuracy for model selection and held-out test accuracy for final evaluation. To assess off-target drift, we evaluate the fine-tuned model on several non-medical benchmarks drawn from the reference evaluation suite. On BBQ [39], we report multiple-choice accuracy as a measure of robustness to social bias. On TruthfulQA [40], we report token-level F1 to capture factual reliability under free-form generation. On IFEval [45], we report prompt accuracy and instruction accuracy, which measure whether the model satisfies explicit formatting and content constraints.

Additional experiments To further show the importance of finetuning in this context, we compare the performance of random selection across various pool sizes. The results are shown in Figure 11. This figure highlights, in particular, that increasing the pool size does not automatically lead to better performance on the target task (here, MedQA), thereby making a case for the use of data selection in this context.

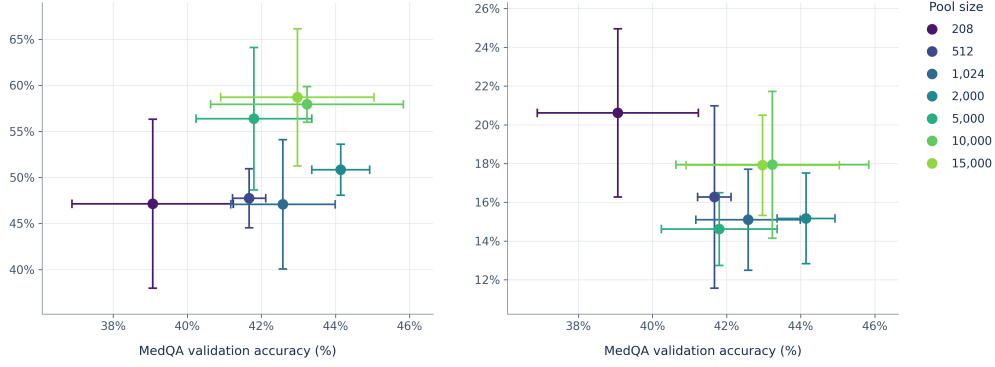


Figure 11: Pool-size effects on MedQA validation accuracy versus BBQ validation accuracy (left) and TruthfulQA validation token F1 (right) Training uses LoRA rank 16, $\alpha = 32$.

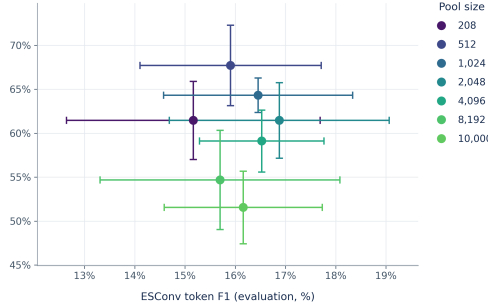


Figure 12: Pool-size effects on ESConv token F1 versus GSM8K validation accuracy. Training uses LoRA rank 64, $\alpha = 128$.

D.3.4 Setting 3: Improving emotional support

We seek to improve emotional-support dialogue performance while leaving the base model’s behavior on mathematical reasoning unchanged. The candidate pool is EmpatheticDialogues [41], a dataset of emotion-grounded open-domain conversations. The target set is ESConv [42], a benchmark of emotional-support conversations. To define the protected reference distribution, we use GSM8K [43], a dataset of grade-school math word problems that probes multi-step mathematical reasoning.

The histograms of ρ_E and ρ_{safe} in Figure 13 describe how the mass of a gradient is distributed across the two subspace. In this setting, most candidate gradients place most of their energy in the residualized target subspace, while the target gradient places a noticeably larger fraction in the reference subspace. This indicates that the working basis does achieve a meaningful separation. This also aligns with the low entanglement score of $\pi_{Q_{\text{ent}}} = 0.08$ in Table 1. Most of the candidate samples that align with the target do not incur the reference-side effect.

Evaluation metrics. We evaluate target-task performance on ESConv using ROUGE-L F1, which is better suited than exact match for this open-ended generation setting. To measure drift away from the reference capability, we report accuracy on GSM8K and additionally on MMLU (zero-shot) as a broader out-of-domain measure of reasoning and knowledge preservation.

Additional experiments and diagnostics. As for setting 2, we compare the performance of random selection across various pool sizes. The results are shown in Figure 12. Again, we observe that increasing the pool size does not automatically lead to better performance on the target task.

We further show the entanglement diagnostics of Section D.2.1.

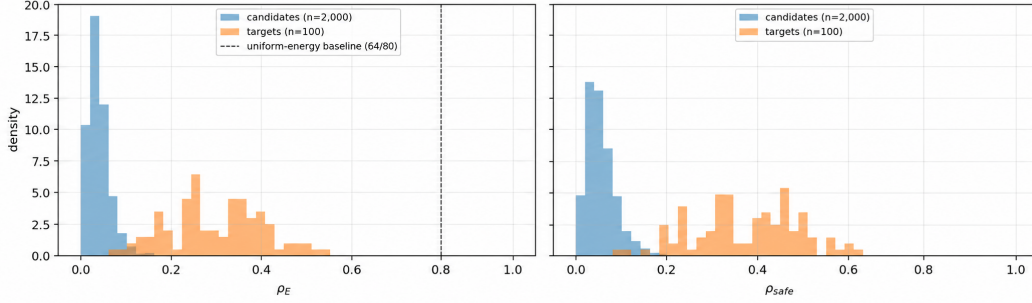


Figure 13: Reference-subspace entanglement diagnostics for Setting 3. We draw the density of ρ_E and ρ_{safe} from (28) with $K_R = 64$, and $K = 80$. Values close to 0 indicate that the gradient lies mostly in the residual task subspace, whereas values close to 1 indicate that it lies mostly in the reference-sensitive subspace.

D.3.5 Setting 4: FLAN instruction tuning

The goal is to improve instruction-following behavior while preserving the base model’s performance on an unrelated knowledge-and-reasoning reference task. This setting is deliberately off-target: the selector is asked to identify useful instruction-tuning examples from a broad FLAN-v2[44] pool using a small IFEval-derived target set, while the reference geometry is estimated from MMLU. Thus, unlike settings where the target, candidate, and reference distributions are closely related, this experiment tests whether a selector can extract target-relevant instruction-following signal without unnecessarily perturbing models’ factual knowledge. The alpha solved for SAFE-DRIFT given (ρ, ϵ) is 2×10^{-4} .

Evaluation metrics. For the target task, we evaluate instruction following on the official IFEval test set [45], which contains 491 prompts. Evaluation is performed with using OLMES [66]. We report the four standard IFEval accuracies: prompt-level strict, prompt-level loose, instruction-level strict, and instruction-level loose. Prompt-level accuracy requires every constraint in a prompt to be satisfied simultaneously, whereas instruction-level accuracy scores each individual constraint separately and averages over constraints. Thus, instruction-level scores are expected to be higher than prompt-level scores. Strict scoring applies the original verifier directly, while loose scoring applies light normalization such as lowercasing, whitespace cleanup, and simple punctuation handling before verification. The OLMES primary scores for IFEval is prompt-level strict and instruction strict accuracy, which we use as the main target metric in Table 2.

For the reference task, we evaluate MMLU on a fixed held-out 200-item test slice. The chat-format mode evaluates MMLU as open-ended generation under the same chat template used during fine-tuning, making it sensitive to whether the fine-tuned model still emits clean single-letter answers such as A, B, C, or D. The loglikelihood mode instead compares

$$P_\theta(A \mid \text{prompt}), P_\theta(B \mid \text{prompt}), P_\theta(C \mid \text{prompt}), P_\theta(D \mid \text{prompt})$$

and selects the answer with maximum probability. This mode is less sensitive to chat-template formatting and provides a cleaner estimate of whether the model’s underlying multiple-choice knowledge has changed. Since the purpose of the reference task is preservation rather than improvement. Values closest to zero indicate the least off-target movement from the base model.

D.4 Ablation Studies

D.4.1 Different selection budgets

We provide the results of Setting 4 with two selection budgets: 5% and 15% of the full candidate pool in Figure 14 for IFEval (target task) and Figure 15 for MMLU (off-target task). Because the target pool is extremely small, with only about 50 IFEval examples, DSIR is particularly brittle in this setting. From Table 1, we know only 22% of candidates fall in Q_{clean} and 26% in Q_{ent} ,

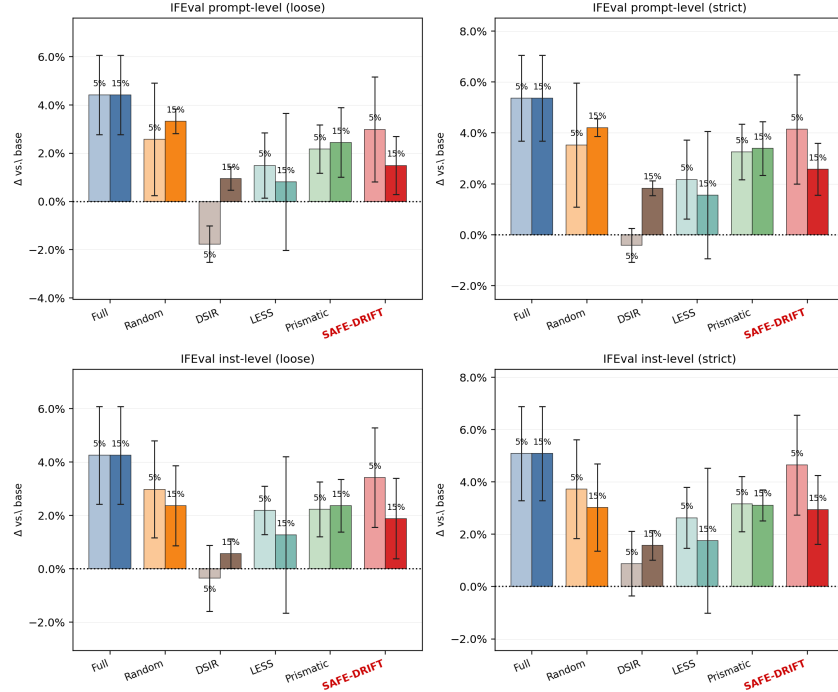


Figure 14: Relative change of IFEval performance compared to the base model. Bar heights are the means and error bars indicate one standard deviation across three seeds. The lighter bar corresponds to a 5% selection budget and the darker bar to a 15% budget.

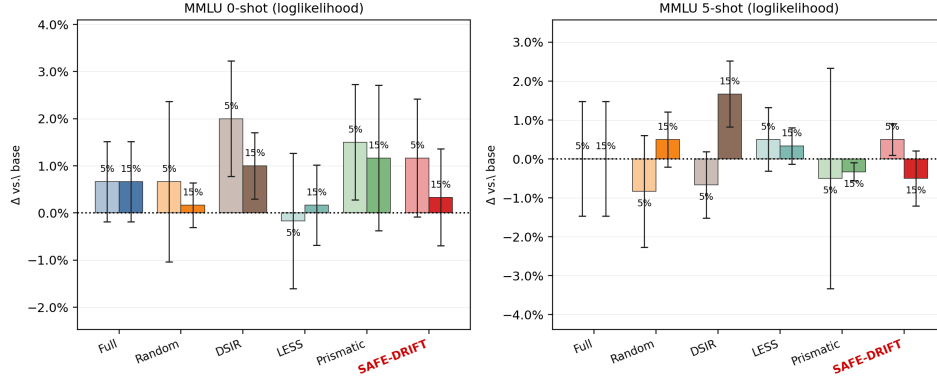


Figure 15: Relative change of MMLU log likelihood of correct answers on the held-out 200 questions compared to the base model: Panels show the 0-shot setting (left) and 5-shot setting (right). The lighter bar corresponds to a 5% selection budget and the darker bar to a 15% budget.

so fewer than half of the FLAN examples have positive target alignment, while the majority are either reference-only or misaligned. Figure 16 also provides evidence that instruction-tuning on the candidate pool and target are substantially entangled: many target-beneficial examples also have positive reference cost (Q_{ent}), so improving IFEval comes at the cost of MMLU-sensitive directions.

Under this setting, SAFE-DRIFT concentrates on the highest-scoring target-beneficial examples at the 5% budget, while at 15%, SAFE-DRIFT selects some weakly target-aligned or reference-entangled examples, diluting the target update and preserving its behavior on the reference task. Comparing with LESS, SAFE-DRIFT gives a more favorable target–reference trade-off and attains more target gains. Random and Prismatic selects a broader subset from the candidate distribution, and they also select many entangled directions that result in MMLU preservation comparable to SAFE-DRIFT.

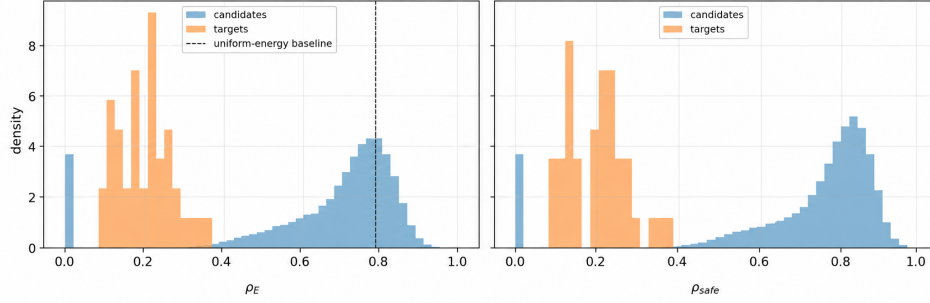


Figure 16: Reference-subspace entanglement diagnostics for Setting 4. We draw the density of ρ_E and ρ_{safe} from (28) with $K_R = 128$, and $K = 160$. The dashed line marks the dimension-ratio baseline $K_R/K = 128/160 = 0.8$; values above this line indicate stronger-than-uniform concentration in reference-sensitive directions.

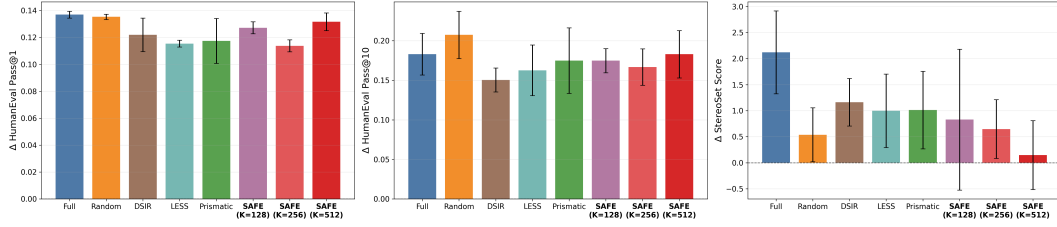


Figure 17: Ablation on the low-rank basis size (K_R, K_T) for SAFE-DRIFT on Setting 1. Left and center panels show change in HumanEval pass@1 and pass@10 relative to the base model; right panel shows change in StereoSet score relative to the base model.

D.4.2 Different choices of rank K_R and K_T

We ablate over three low-rank settings $(K_R, K_T) \in \{(128, 32), (256, 64), (512, 128)\}$ for SAFE-DRIFT on Setting 1, with results shown in Figure 17 for selection budget 15%. SAFE-DRIFT is robust to the choice of (K_R, K_T) : all three variants achieve competitive HumanEval pass@1 and pass@10 rates, and all produce smaller changes in StereoSet score than most baselines. One thing to note is that the basis $(K_R, K_T) = (512, 128)$ achieves the smallest off-target drift. This suggests that a larger basis can be beneficial when more candidates are available, but the gains are modest and all three settings consistently outperform the baselines on the reference metric.

Figure 18 further shows the cumulative reference drift D_R and cumulative validation gain over training steps for all methods at both budgets. SAFE-DRIFT variants achieve lower cumulative reference cost at $(K_R, K_T) = (128, 32), (512, 128)$ than baselines at comparable or higher validation gain, confirming that the selection criterion successfully trades reference cost for target gain rather than optimizing either in isolation.

D.4.3 Using reference-induced norm vs Euclidean norm in the selector.

We evaluate the effect of using the reference Fisher-induced norm and the Euclidean norm in the data selection objective in Eq. (7). We present results for the MedQA and Esconv datasets corresponding to Setting 2 and 3 (see Table 1). We report metrics on the target and reference tasks with varying off-target drifts ρ (Figures 21 and 19), and visualize the selected gradients (Figures 22 and 20). Overall, we see that selecting data points in the Euclidean geometry yields worse performance in controlling the off-target grid: we consistently see that, in the Euclidean geometry, selected gradients have high projections along the reference geometry, irrespective of the value of ρ . By contrast, selecting points in the reference geometry yields a stronger sensitivity to higher off-target costs.

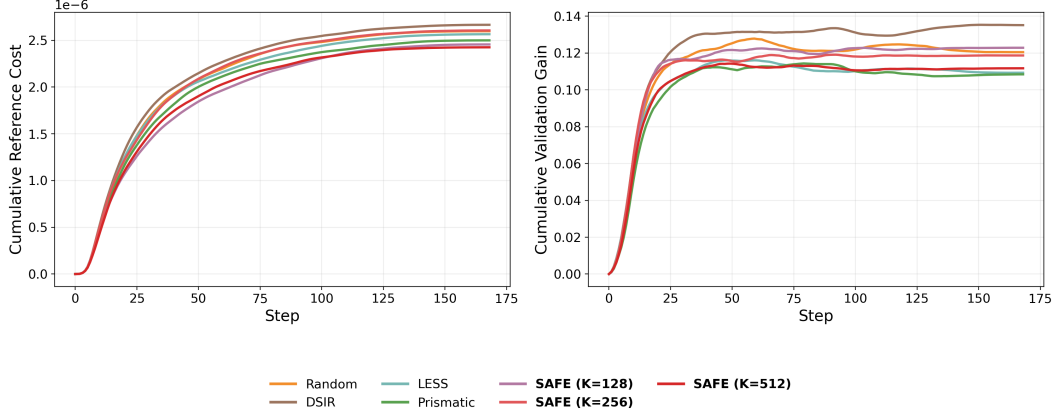


Figure 18: Cumulative reference drift, $D_{\mathcal{R}} = \Delta\theta^\top \hat{\mathbf{F}}_R^{(K_R)} \Delta\theta$ and cumulative validation gain $-\Delta\theta^\top \hat{b}_{\text{tar}}^{(K)}$ over training steps.

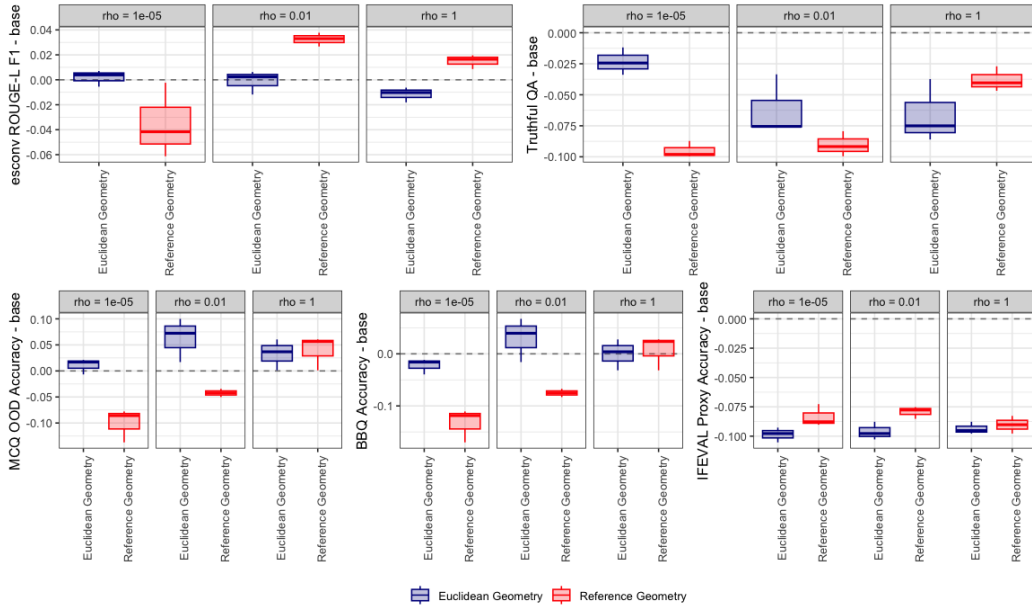


Figure 19: MedQA experiment (Setting 2): Geometry. Why is there no MedQA accuracy reported here? Also, there are added reference datasets—need explanation.

D.4.4 Effect of the optimizer preconditioner

We evaluate the effect of the optimizer-induced preconditioner $\mathbf{P}$. Our framework allows different choices of $\mathbf{P}$, including the identity preconditioner corresponding to SGD-style updates and an Adam-style diagonal preconditioner estimated from warm-up gradients. This ablation tests whether the observed behavior of SAFE-DRIFT depends critically on using the Adam preconditioner.

Figure 3 and figures 23-25 show that the best preconditioner varies across settings. Adam performs better in Setting 3, while SGD is more conservative on several Setting 2 reference metrics. This is expected because the preconditioner changes the effective update induced by each selected example. Importantly, both variants retain the same qualitative target-reference trade-off, suggesting that SAFE-DRIFT is not tied to a single optimizer approximation. In practice, we recommend using the preconditioner that matches the downstream fine-tuning optimizer.

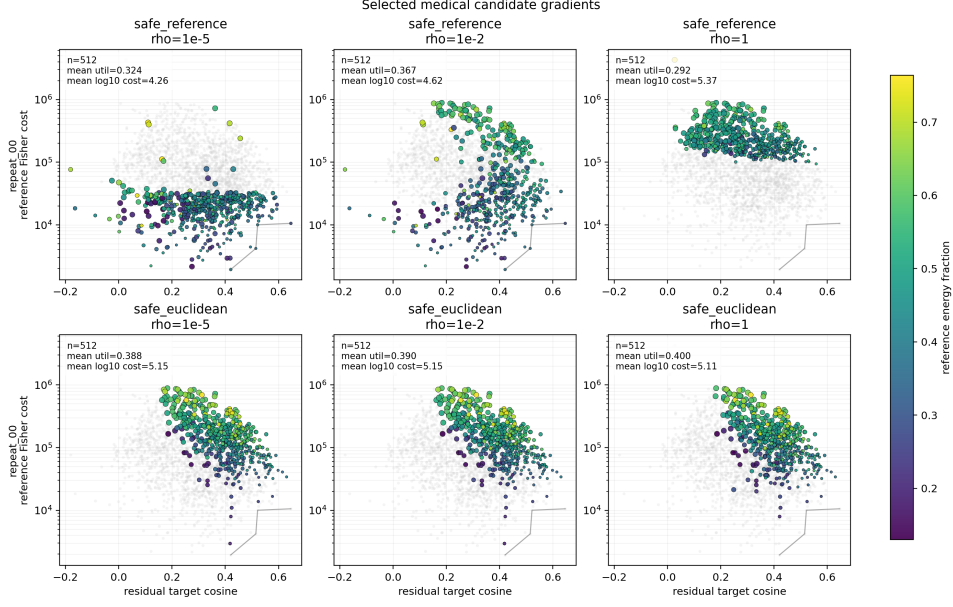


Figure 20: MedQA experiment (Setting 2): Selected Gradients.

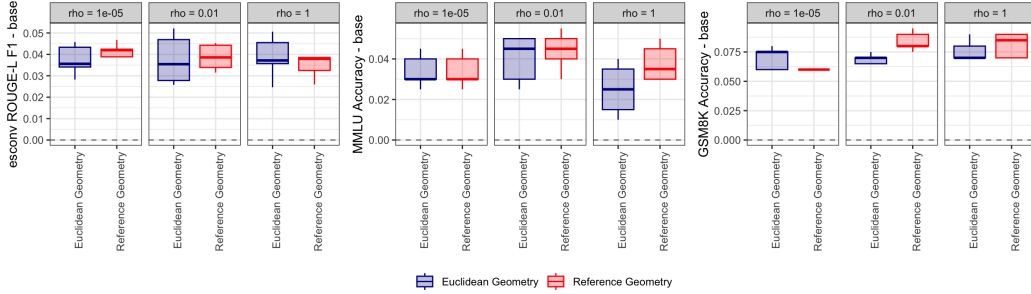


Figure 21: Esconv experiment (Setting 3): Geometry.

D.5 Comparison Against Behavior-Preservation Baselines

[inline]Add comment on table 3 We provide the complementary results of replay-based methods of DSIR, LESS, and Prismatic of Section 3.1 in Table 3. We additionally experiment with augmenting the reference KL-divergence penalty to the SFT loss:

$$\mathcal{L}_{\text{SFT}}(\theta) + \lambda \mathbb{E}_{x \sim R} \left[\frac{1}{|x|} \sum_t D_{\text{KL}}(p_{\theta_0}(\cdot | x_{<t}) \| p_{\theta}(\cdot | x_{<t})) \right],$$

and sweep over $\lambda \in [0, 0.01, 0.1, 1]$ and select the best λ on a validation set using the geometric mean of target and reference performance.

Rehearsal moves every baseline in the wrong direction on both axes as the replay fraction grows from 10% to 20%: target performance falls while realized off-target drift rises. The KL penalty behaves as intended—across positive λ , the Spearman correlation with held-out KL D_{KL} is -0.949 and -0.896 for LESS and Prismatic in Setting 2, and -0.949 for both in Setting 3—but stronger regularization generally reduces or fails to improve target accuracy. In Setting 2 regularized Prismatic preserves reference accuracy more effectively while SAFE-DRIFT more than halves the reference drift of DSIR at comparable target gain, so the two occupy different points on the same trade-off rather than one dominating the other. In Setting 3 validation selects $\lambda = 0$ for both methods, consistent with that setting’s low entanglement, and SAFE-DRIFT achieves better and more stable reference accuracy.

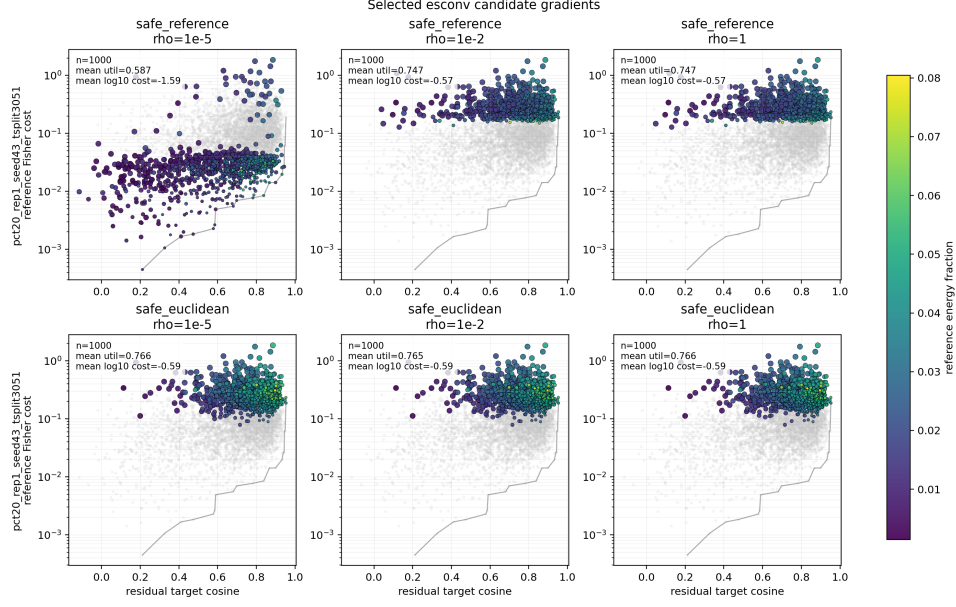


Figure 22: Esconv experiment (Setting 3): Selected Gradients.

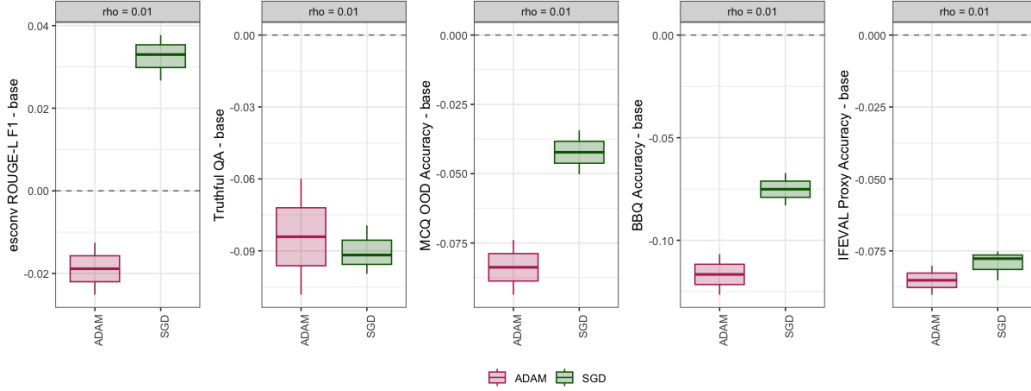


Figure 23: MedQA experiment (Setting 2), Why is there no MedQA accuracy reported here? Also, there are added reference datasets—need explanation.

D.6 Robustness Across Base Models

We repeat Setting 1 across two model families and five configurations—Qwen2.5 at 1.5B, 3B and 7B, and OLMo-2 at 7B and 13B—at a 15% selection budget with three seeds. Table 4 reports the full numbers summarized in Figure 2(a–c), including fine-tuning on the full candidate pool. SAFE-DRIFT attains the lowest predicted drift $D_{\mathcal{R}}$ in every configuration, reducing $D_{\mathcal{R}}$ by 10.2–18.0% and D_{KL} by 31.1–60.0% relative to full-data fine-tuning. Realized drift D_{KL} is less uniform: SAFE-DRIFT is lowest at Qwen2.5-1.5B and OLMo-2-13B, and competitive but not best elsewhere. Target performance shows a similar split, with SAFE-DRIFT leading on the smaller Qwen models but not the larger ones—trading target gain for drift control where the target directions are entangled with the reference geometry.

D.7 Choosing the Drift Budget ρ

The drift budget ρ is the main user-facing control in SAFE-DRIFT: tighter budgets avoid high-cost reference directions, looser ones let the selector pursue target gain. Section 3.1 sweeps ρ over $\{10^{-5}, 10^{-2}, 10^0\}$ in Setting 2; Table 5 reports rank correlations over a larger sweep of nine

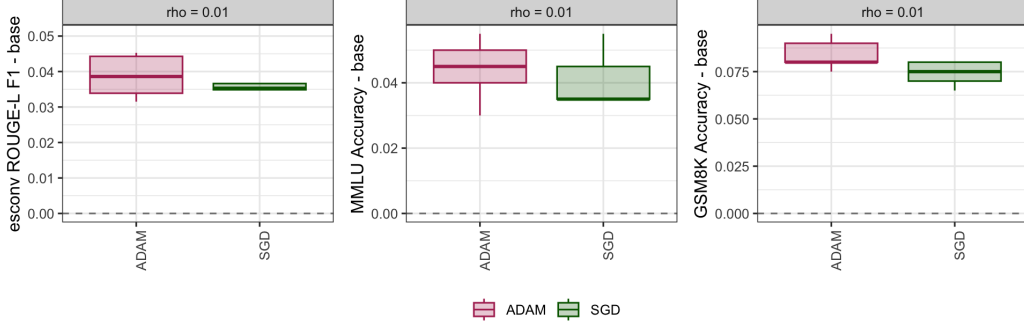


Figure 24: Esconv experiment (Setting 3): Selected Gradients.

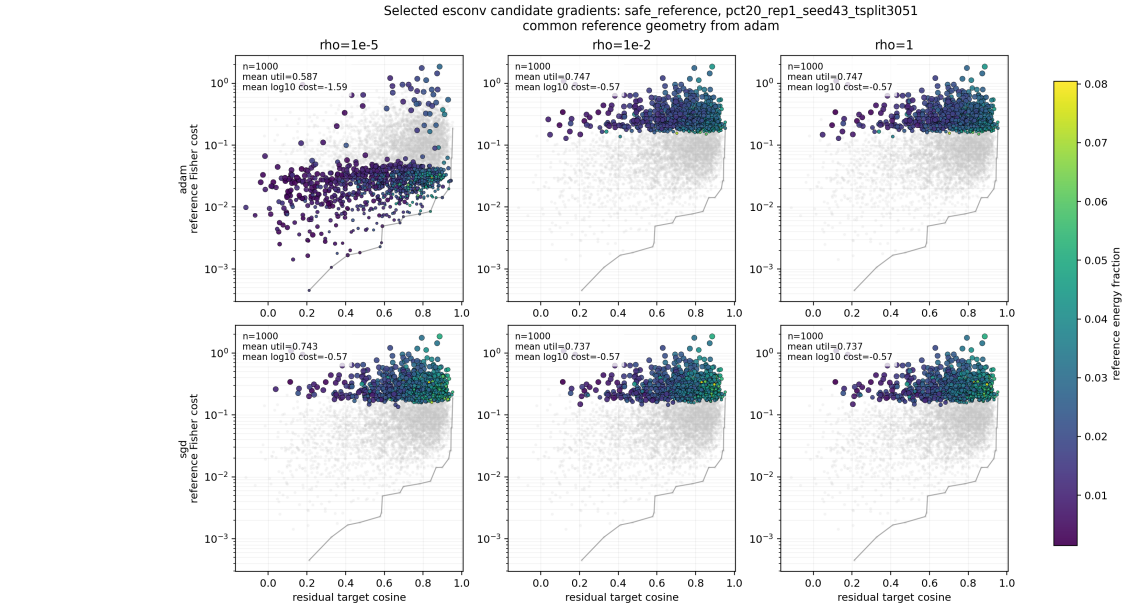


Figure 25: Esconv experiment (Setting 3): Selected Gradients.

hyperparameter configurations in Setting 2 and eighteen in Setting 3, five seeds each. Increasing ρ reliably raises target accuracy (Spearman correlation 0.753 and 0.672), but its association with reference drift reverses sign between settings: 0.483 in Setting 2, where a looser budget costs reference accuracy, against -0.735 in Setting 3, where it improves both axes at once (Figure 26). ρ is thus a trade-off parameter, not a universally monotone regularizer. The sweep also confirms that the surrogate the selector constrains tracks the drift actually incurred as the budget moves: $D_{\mathcal{R}}$ and D_{KL} correlate at $\rho_s = 0.967$ and 0.670 .

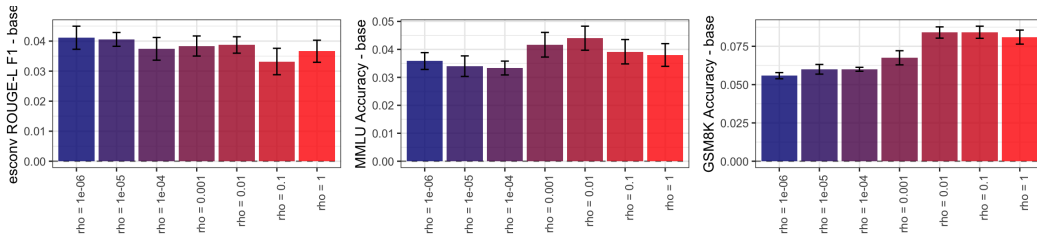


Figure 26: Effect of ρ in ESconv (Setting 3)

(a) Rehearsal (replay) baselines — Setting 1					
Method	Replay %	Δ HumanEval@1 $\uparrow$	Δ HumanEval@10 $\uparrow$	Held-out KL $\downarrow$	
DSIR	10	0.0516 $\pm$ 0.0096	0.1362 $\pm$ 0.0040	0.0190 $\pm$ 0.0030	
DSIR	20	0.0429 $\pm$ 0.0108	0.1260 $\pm$ 0.0090	0.0223 $\pm$ 0.0007	
LESS	10	0.0217 $\pm$ 0.0056	0.0870 $\pm$ 0.0090	0.0152 $\pm$ 0.0017	
LESS	20	0.0243 $\pm$ 0.0082	0.0995 $\pm$ 0.0153	0.0181 $\pm$ 0.0028	
Prismatic	10	0.0185 $\pm$ 0.0009	0.0793 $\pm$ 0.0060	0.0197 $\pm$ 0.0020	
Prismatic	20	0.0134 $\pm$ 0.0038	0.0750 $\pm$ 0.0130	0.0230 $\pm$ 0.0040	
SAFE-DRIFT	0	0.115 $\pm$ 0.004	0.264 $\pm$ 0.031	0.0127 $\pm$ 0.0006	
(b) KL-regularized selection baselines — Settings 2 and 3					
Setting	Method	Selected λ	Δ test target $\uparrow$	Δ test reference $\uparrow$	Eval KL $\downarrow$
2	DSIR	—	+3.59 $\pm$ 2.04	−8.10 $\pm$ 3.58	4.210 $\pm$ 0.528
	LESS	$\lambda = 0.1$	+2.08 $\pm$ 0.60	+0.73 $\pm$ 1.26	1.974 $\pm$ 0.224
	Prismatic	$\lambda = 1$	+2.99 $\pm$ 1.63	+0.89 $\pm$ 2.07	1.534 $\pm$ 0.025
	SAFE-DRIFT	—	+3.36 $\pm$ 1.42	−3.50 $\pm$ 3.24	4.003 $\pm$ 0.467
3	DSIR	—	+4.65 $\pm$ 1.18	+57.97 $\pm$ 7.15	0.317 $\pm$ 0.025
	LESS	$\lambda = 0$	+ 6.77 $\pm$ 1.18	+64.58 $\pm$ 7.59	0.368 $\pm$ 0.039
	Prismatic	$\lambda = 0$	+4.55 $\pm$ 0.97	+61.46 $\pm$ 3.85	0.364 $\pm$ 0.023
	SAFE-DRIFT	—	+4.51 $\pm$ 1.10	+ 65.16 $\pm$ 2.63	0.391 $\pm$ 0.011

Table 3: *Comparison against behavior-preservation baselines.* (a) Rehearsal: reference examples, drawn from the Fisher-estimation split and disjoint from the drift-evaluation set, are mixed into training at 10% and 20% of the budget. SAFE-DRIFT never trains on the reference set. (b) LESS and Prismatic augmented with a reference KL penalty with λ chosen on validation set by the geometric mean of target and reference accuracy. We show means and standard deviations of differences from each setting’s base-model. **To me, the benefits of SAFE-DRIFT compared to baselines are not clear here. The reference side metrics are generally worse. What do "reference side metrics" mean?**

Model	Evaluation	Random	DSIR	LESS	Prismatic	SAFE-DRIFT	Full Data
Qwen2.5-1.5B	HumanEval pass@1 $\uparrow$	0.128 $\pm$ 0.023	0.133 $\pm$ 0.003	0.157 $\pm$ 0.083	0.165 $\pm$ 0.018	0.283 $\pm$ 0.020	0.327 $\pm$ 0.010
	Ref. Fisher cost $\downarrow$	−5.58 $\pm$ 0.04	−5.06 $\pm$ 0.06	−5.48 $\pm$ 0.06	−5.64 $\pm$ 0.04	−6.29 $\pm$ 0.04	−5.60 $\pm$ 0.05
	Held-out ref. KL $\downarrow$	0.0049 $\pm$ 0.0001	0.0055 $\pm$ 0.0006	0.0057 $\pm$ 0.0005	0.0044 $\pm$ 0.0001	0.0044 $\pm$ 0.0003	0.0110 $\pm$ 0.0001
Qwen2.5-3B	HumanEval pass@1 $\uparrow$	0.162 $\pm$ 0.028	0.175 $\pm$ 0.032	0.193 $\pm$ 0.017	0.129 $\pm$ 0.049	0.242 $\pm$ 0.053	0.248 $\pm$ 0.026
	Ref. Fisher cost $\downarrow$	−6.16 $\pm$ 0.03	−5.65 $\pm$ 0.06	−6.13 $\pm$ 0.03	−6.22 $\pm$ 0.01	−6.89 $\pm$ 0.03	−6.25 $\pm$ 0.06
	Held-out ref. KL $\downarrow$	0.0031 $\pm$ 0.0003	0.0038 $\pm$ 0.0004	0.0042 $\pm$ 0.0008	0.0031 $\pm$ 0.0001	0.0036 $\pm$ 0.0002	0.0091 $\pm$ 0.0008
Qwen2.5-7B	HumanEval pass@1 $\uparrow$	0.117 $\pm$ 0.041	0.089 $\pm$ 0.045	0.180 $\pm$ 0.055	0.103 $\pm$ 0.053	0.159 $\pm$ 0.044	0.269 $\pm$ 0.040
	Ref. Fisher cost $\downarrow$	−6.89 $\pm$ 0.02	−6.41 $\pm$ 0.04	−6.81 $\pm$ 0.02	−6.89 $\pm$ 0.02	−7.62 $\pm$ 0.03	−6.88 $\pm$ 0.13
	Held-out ref. KL $\downarrow$	0.0038 $\pm$ 0.0003	0.0038 $\pm$ 0.0002	0.0039 $\pm$ 0.0006	0.0030 $\pm$ 0.0004	0.0037 $\pm$ 0.0001	0.0076 $\pm$ 0.0010
OLMo-2-7B	HumanEval pass@1 $\uparrow$	0.130 $\pm$ 0.023	0.148 $\pm$ 0.021	0.117 $\pm$ 0.026	0.135 $\pm$ 0.013	0.125 $\pm$ 0.033	0.123 $\pm$ 0.010
	Ref. Fisher cost $\downarrow$	−5.98 $\pm$ 0.08	−5.38 $\pm$ 0.06	−5.72 $\pm$ 0.13	−5.90 $\pm$ 0.08	−7.05 $\pm$ 0.15	−5.95 $\pm$ 0.07
	Held-out ref. KL $\downarrow$	0.0106 $\pm$ 0.0021	0.0110 $\pm$ 0.0005	0.0098 $\pm$ 0.0006	0.0129 $\pm$ 0.0031	0.0127 $\pm$ 0.0006	0.0209 $\pm$ 0.0021
OLMo-2-13B	HumanEval pass@1 $\uparrow$	0.180 $\pm$ 0.043	0.182 $\pm$ 0.029	0.154 $\pm$ 0.025	0.194 $\pm$ 0.036	0.161 $\pm$ 0.031	0.185 $\pm$ 0.029
	Ref. Fisher cost $\downarrow$	−6.88 $\pm$ 0.17	−6.23 $\pm$ 0.08	−6.71 $\pm$ 0.06	−6.69 $\pm$ 0.10	−7.95 $\pm$ 0.07	−6.83 $\pm$ 0.07
	Held-out ref. KL $\downarrow$	0.0097 $\pm$ 0.0013	0.0097 $\pm$ 0.0027	0.0087 $\pm$ 0.0011	0.0095 $\pm$ 0.0010	0.0084 $\pm$ 0.0013	0.0122 $\pm$ 0.0012

Table 4: *Base model ablation* (Setting 1). Two model families and five configurations; 15% selection budget and following configurations in Table 1. We measure reference Fisher cost $D_{\mathcal{R}}$ and the held-out reference KL D_{KL} , each corresponding to predicted and realized off-target drift. Bold denotes the best selection method per row; “Full Data” is shown for reference and not ranked.

This suggests two ways to set ρ . Under a *fixed* budget it is read directly as a rough upper bound on the increase in probability of any undesirable output, and chosen from the drift the practitioner is willing to accept. Under a *data-driven* budget it is tuned as an ordinary hyperparameter on validation target and drift sets, revealing what a given ρ actually buys on that dataset and optimizer. We recommend the latter whenever a validation drift set is available, since the entanglement of the setting—not the nominal value of ρ —determines the cost of the budget.

Setting	Optimizer	Hyperparam. configs	Seeds per cell	ρ vs. target	ρ vs. ref. drift	Target vs. ref. drift	$D_{\mathcal{R}}$ vs. D_{KL}
Setting 2	Adam	9	5	0.753	0.483	0.561	0.967
Setting 3	Adam	18	5	0.672	-0.735	-0.820	0.670

Table 5: *Spearman rank correlations for the drift budget ρ* . Reference drift in the ρ and target columns is D_{task} , the difference in reference task accuracy between the base and fine-tuned model. The negative Setting 3 correlations indicate that target and reference accuracy increase together—consistent with the low target–reference entanglement of that setting. The final column correlates the predicted drift $D_{\mathcal{R}}$ with the realized drift D_{KL} over the same runs.

D.8 Composite Reference Distributions

Composition	Method	Δ Target	Δ B	Δ T	Δ K	Δ Ref GM	Δ All-ref GM	Δ KL $\downarrow$
B+T	SAFE-DRIFT	+0.090	+0.085	-0.105	+0.070	-0.062	-0.032	5.17
	Random	+0.063	-0.049	-0.107	+0.061	-0.100	-0.065	5.47
	DSIR	+0.064	-0.054	-0.125	+0.078	-0.117	-0.078	5.49
	LESS	+0.022	+0.028	-0.100	+0.033	-0.072	-0.049	5.54
	Prismatic	+0.078	-0.042	-0.107	+0.055	-0.098	-0.065	5.46
B+K	SAFE-DRIFT	+0.086	+0.066	-0.093	+0.051	+0.058	-0.031	5.16
	Random	+0.063	-0.049	-0.107	+0.061	+0.004	-0.065	5.47
	DSIR	+0.064	-0.054	-0.125	+0.078	+0.009	-0.078	5.49
	LESS	+0.022	+0.028	-0.100	+0.033	+0.030	-0.049	5.54
	Prismatic	+0.078	-0.042	-0.107	+0.055	+0.005	-0.065	5.46
T+K	SAFE-DRIFT	+0.094	+0.046	-0.103	+0.063	-0.066	-0.041	5.29
	Random	+0.063	-0.049	-0.107	+0.061	-0.069	-0.065	5.47
	DSIR	+0.064	-0.054	-0.125	+0.078	-0.083	-0.078	5.49
	LESS	+0.022	+0.028	-0.100	+0.033	-0.070	-0.049	5.54
	Prismatic	+0.078	-0.042	-0.107	+0.055	-0.071	-0.065	5.46
B+T+K	SAFE-DRIFT	+0.078	+0.038	-0.098	+0.020	-0.048	-0.048	5.13
	Random	+0.063	-0.049	-0.107	+0.061	-0.065	-0.065	5.47
	DSIR	+0.064	-0.054	-0.125	+0.078	-0.078	-0.078	5.49
	LESS	+0.022	+0.028	-0.100	+0.033	-0.049	-0.049	5.54
	Prismatic	+0.078	-0.042	-0.107	+0.055	-0.065	-0.065	5.46

Table 6: *Composite reference distributions* (Setting 2). B = BBQ, T = TruthfulQA, K = MMLU (non-medical knowledge). Each composition names the reference tasks used to construct the SAFE-DRIFT constraint; omitted tasks are still evaluated but not protected. All compositions optimize the same target (MedQA accuracy). Δ Ref GM is the geometric mean of accuracies over the *protected* reference tasks; Δ All-ref GM is the geometric mean over all three. **Why are the values for Random, DSIR, LESS, Prismatic all the same across composition? Because the mixture characterizes the reference distribution. There is no reference in DSIR, etc**

Question—see Table 6 SAFE-DRIFT constrains drift on a reference distribution that the practitioner specifies, so it is safe with respect to the failure modes named in that reference rather than universally safe. A natural question is whether the method degrades when the reference is made more complex. We therefore rerun Setting 2 with the reference constructed from several benchmarks at once: writing B for BBQ, T for TruthfulQA and K for MMLU (non-medical knowledge), we protect B+T, B+K, T+K and all three jointly. Every composition optimizes the same target (MedQA accuracy); benchmarks omitted from a composition are still evaluated but not protected.

Table 6 reports the results. Across all four compositions SAFE-DRIFT attains the lowest drift on the protected tasks and the lowest reference KL, and performance is not smothered as the reference grows: the target gain falls only from +0.090 under the smallest composition to +0.078 under the largest. The gain also transfers to reference tasks that are *not* protected, as measured by the geometric mean over all three benchmarks, suggesting that the reference Fisher geometry estimated from a subset of benchmarks captures directions shared with the others.

Setting	Block	1	8	32	64	128	256	512	1024
Setting 1	Reference	0.133	0.257	0.414	0.525	0.654	0.794	0.924	1.000
	Task/candidate	0.074	0.258	0.488	0.627	0.748	0.823	0.885	0.937
Setting 3	Reference	0.234	0.443	0.685	0.847	1.000	—	—	—
	Task/candidate	0.118	0.279	0.448	0.564	0.704	0.862	0.9999	—

Table 7: *Cumulative spectral energy* (squared singular-value energy) as a function of rank, mean over three seeds. The approximation error of the top- K truncation of the reference Fisher and target gradient is controlled by the spectral tail, so these curves bound the missing reference cost and are necessarily dataset dependent.

Update	Test MedQA accuracy	Test reference accuracy
LoRA	44.30% $\pm$ 1.42%	36.76% $\pm$ 3.58%
Full SFT	37.50% $\pm$ 3.87%	38.92% $\pm$ 2.42%

Table 8: *LoRA versus full supervised fine-tuning* (Setting 2).

D.9 Quality of the Low-Rank Approximation

The reference Fisher and target gradient are represented in a shared low-rank basis, which is used only to score and select candidates—it does not constrain the fine-tuning update itself. Its approximation error is governed by the spectral tail: the missing reference cost after truncating $\hat{\mathbf{F}}_{\mathcal{R}}$ at rank K_R is bounded by the energy discarded, so the quality of the approximation is dataset dependent.

Table 7 reports cumulative squared singular-value energy as a function of rank, averaged over three seeds. At the ranks used in the paper, Setting 1 retains 92.4% of reference energy at $K_R = 512$ and 74.8% of task energy at $K_T = 128$. In Setting 3 the target is computed over 128 reference and 512 task/candidate examples, which is why the curves plateau at those ranks. Results are stable to the truncation: across hyperparameter values in Setting 3, the Pearson correlation between results at the two rank settings we tested is 0.978 over seed-averaged means, and Appendix D.4.2 shows the same stability in Setting 1. Consistent with the synthetic study of Section ??, tasks with flatter gradient spectra require substantially larger rank, while tasks whose spectrum concentrates on a few directions are almost unaffected by the truncation.

D.10 LoRA Versus Full Supervised Fine-Tuning

LoRA defines the trainable parameter space throughout our experiments, because it has become the dominant approach to practical parameter-efficient instruction tuning. There is no conceptual obstacle to running SAFE-DRIFT over a full SFT parameterization—only a computational one, since per-example gradients must then be formed in the full parameter space.

Table 8 compares the two on Setting 2. The LoRA parameterization yields higher target accuracy while the full update controls the reference distribution better, which is consistent with the larger update subspace giving the drift constraint more directions in which to place the update.

D.11 Computational Cost

SAFE-DRIFT has two computational bottlenecks. The first is per-example gradient construction over the candidate pool, which scales linearly in N_C and is equally required by LESS and Prismatic; the second is the low-rank estimation of the reference Fisher and target gradient, whose marginal cost is small by comparison. For large candidate pools the pool can be subsampled before selection, and for large models LoRA keeps the per-example gradients tractable.

Table 9 reports instrumented FLOP counts for every method and setting. Because the gradient construction is shared, SAFE-DRIFT incurs *identical* selection FLOPs to LESS and Prismatic; the drift constraint itself is not what makes gradient-based selection expensive. For all gradient-based selectors, selection rather than training dominates total compute, accounting for 82.7–97.6% of total FLOPs across settings.

Setting	Model	Method	Select FLOPs	Train FLOPs	Total FLOPs	Select share	Runs
Setting 1	OLMo-2-7B	Random	0	7.24×10^{15}	7.24×10^{15}	0.0%	3
		DSIR	$< 10^8$	4.13×10^{15}	4.13×10^{15}	$< 0.1\%$	3
		LESS	5.54×10^{16}	6.05×10^{15}	6.14×10^{16}	90.2%	3
		Prismatic	5.54×10^{16}	7.74×10^{15}	6.31×10^{16}	87.7%	3
		SAFE-DRIFT	5.54×10^{16}	1.16×10^{16}	6.70×10^{16}	82.7%	3
	OLMo-2-13B	Random	0	1.39×10^{16}	1.39×10^{16}	0.0%	3
		DSIR	$< 10^8$	7.92×10^{15}	7.92×10^{15}	$< 0.1\%$	3
		LESS	1.06×10^{17}	1.30×10^{16}	1.19×10^{17}	89.1%	3
		Prismatic	1.06×10^{17}	1.48×10^{16}	1.21×10^{17}	87.8%	3
		SAFE-DRIFT	1.06×10^{17}	2.19×10^{16}	1.28×10^{17}	82.9%	3
Setting 2	OLMo-2-7B-Instruct	Random	0	3.056×10^{15}	1.122×10^{16}	0.0%	5
		DSIR	4.865×10^7	3.560×10^{15}	1.172×10^{16}	$< 0.1\%$	5
		LESS	1.219×10^{17}	3.568×10^{15}	1.337×10^{17}	91.2%	5
		Prismatic	1.219×10^{17}	3.035×10^{15}	1.331×10^{17}	91.6%	5
		SAFE-DRIFT	1.219×10^{17}	1.625×10^{15}	1.281×10^{17}	95.2%	5
Setting 3	OLMo-2-7B-Instruct	Random	0	4.420×10^{15}	1.586×10^{16}	0.0%	5
		DSIR	1.054×10^8	2.951×10^{15}	1.439×10^{16}	$< 0.1\%$	5
		LESS	4.988×10^{17}	5.405×10^{15}	5.156×10^{17}	96.7%	5
		Prismatic	4.988×10^{17}	4.371×10^{15}	5.161×10^{17}	96.6%	5
		SAFE-DRIFT	4.988×10^{17}	3.006×10^{15}	5.108×10^{17}	97.6%	5

Table 9: *Full compute analysis*. All runs use Adam with LoRA. “Select share” is the fraction of total FLOPs spent on data selection.