LEARNING TRANSFERABLE DIRECTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Transferability need not be a property of an entire source model. A source update that is globally misaligned with a target can contain locally useful computations, while a globally similar source can move particular target inputs in a harmful direction. We introduce *Learning Transferable Directions* (LTD), which represents each reusable source block by the exact function-space change it induces at a target input and learns a target loss-descent direction field from cross-fitted target gradients. A cheap proposal network first shortlists source blocks; only those blocks are evaluated, after which a sparse nonnegative quadratic program composes locally aligned directions. The construction requires source checkpoints but not source examples and nests target-only prediction exactly. We establish a smoothness-based directional-gain bound, exhibit a partial-sharing setting in which every input-independent source-weight rule incurs a nonvanishing gap, and give an optional finite-sample validation gate that falls back to target-only prediction unless transfer is certified nonharmful for a prespecified bounded loss. Simulations and real-data experiments show that LTD improves target performance in low-resource regimes, reduces negative transfer, and achieves favorable performance-computation tradeoffs over model-merging and routing baselines across vision and language benchmarks.

1 INTRODUCTION

Many transfer-learning methods effectively ask whether a source model is sufficiently compatible with a target to be reused. Compatibility is then expressed through pooling, fine-tuning, source-level weighting, model merging, or constraints on parameter geometry (Pfeiffer et al., 2021; Wortsman et al., 2022; Ilharco et al., 2023; Yang et al., 2024b; Gu et al., 2025). This source-level view can be too coarse. A fine-tuned model may contain a broadly useful representation change, a task-specific correction, and a shortcut that reverses under target shift. More importantly, the same source update can help one target input and hurt another. The relevant question is therefore *which source computation points in a useful target direction for this input?* Figure 1 illustrates the resulting geometry; a source computation can align with the target descent direction in one region of the input space while opposing it in another, so transferability can be local, not reducible to a single property of the source model.

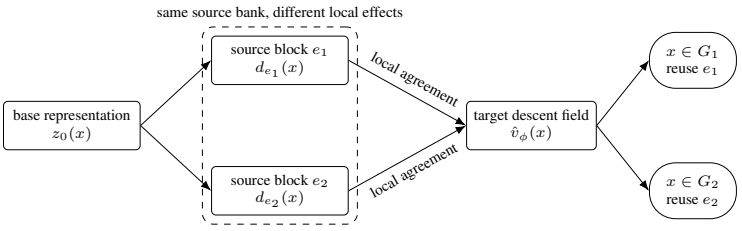


Figure 1: **Transferability as local directional agreement.** Here $z_0(x)$ denotes the base representation, e_j a reusable source block, $d_{e_j}(x)$ its induced function-space change, and $\hat{v}_\phi(x)$ the learned target loss-descent field; G_1 and G_2 denote target regions. For target input x, transferability is assessed by directional agreement between $d_{e_j}(x)$ and $\hat{v}_\phi(x)$. Different target regions can therefore favor different source computations, even when input-independent source weights obscure opposing local effects.

Angle-based transfer provides a useful starting point. In high-dimensional regression, source and target parameters can be far apart in Euclidean distance while remaining directionally concordant, and angular information can mitigate negative transfer when distance is misleading (Gu et al., 2025). Deep model reuse, however, involves richer geometries. Weight-space methods combine fine-tuning updates

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

or reduce interference among them through averaging, curvature information, sparsification, or adaptive task weights (Wortsman et al., 2022; Ilharco et al., 2023; Matena and Raffel, 2022; Yadav et al., 2023; Yu et al., 2024; Yang et al., 2024b). Localization methods identify task-relevant parameter subsets or correct representations after merging (Wang et al., 2024; Yang et al., 2024a; He et al., 2024; Yan et al., 2025), while input-dependent expert methods learn which modules or experts to combine (Chen et al., 2024; Feng et al., 2024; Lu et al., 2024; Wang et al., 2024; Muqeeth et al., 2024). Transferability scores instead assess whether a pretrained representation is promising for a target task (Achille et al., 2019; Nguyen et al., 2020; You et al., 2021; Huang et al., 2022). These approaches localize, score, or combine parameters, models, representations, or experts. LTD uses a different transfer object: the *exact function-space change induced by a reusable source block on a particular target input*, evaluated against the target loss-descent field at that input. Input-dependent routing alone is therefore not the distinction; the transfer decision is driven by the measured local functional effect of a source computation and its directional agreement with target descent. Two source blocks with similar parameter geometry can consequently induce opposite target-space directions on the same input.

This mismatch between parameter and functional geometry is not incidental. Neural networks admit parameter symmetries and connected low-loss regions, while the apparent similarity of learned representations depends on the geometry used to compare their activations (Kornblith et al., 2019; Frankle et al., 2020; Ferbach et al., 2024). Negative-transfer studies further show that reusing a source can degrade target performance and that different spectral or representational components need not transfer uniformly (Chen et al., 2019; Wang et al., 2019). These observations motivate evaluating transfer where its consequence is expressed, i.e., in the target function, without relying on a global parameter metric to infer local usefulness. Accordingly, LTD formulates transferability through input-dependent functional alignment between source-induced changes and the target descent direction.

Our contributions are fourfold. First, we define a transferable direction as an input-dependent functional change and show that, in linear models, global parameter angles arise as a second-moment-weighted special case. Second, we develop a checkpoint-only transfer algorithm that requires the base and source checkpoints but no source examples; a cheap proposal stage shortlists candidate blocks, exact functional directions are evaluated only for that shortlist, and a sparse nonnegative quadratic program composes the resulting correction. Third, we establish complementary theoretical guarantees: a directional-gain bound, a partial-sharing construction in which every input-independent source-weight rule incurs a nonvanishing gap relative to local routing, and a finite-sample target-only fallback guarantee for an optional validation gate. Fourth, simulations and real-data experiments evaluate both predictive performance and the proposed geometric mechanism through low-resource adaptation, model-merging benchmarks, negative-transfer diagnostics, and interventions on selected source directions.

2 Partial Transfer and Local Functional Geometry

We now formalize transfer at the level of input-dependent functional changes. For an integer m, let $[m] = \{1, \dots, m\}$, and let $\langle u, v\rangle$ and $\|u\|_2$ denote the Euclidean inner product and norm. We observe a labeled target sample $\mathcal{D}_0 = \{(X_i, Y_i)\}_{i=1}^{n_0}$ from target distribution P_0 and have access to a common base checkpoint θ_0 and S source checkpoints $\{\theta_s : s \in [S]\}$. We write $\Delta_s = \theta_s - \theta_0$ for the update associated with source s. Source examples are not required at target adaptation time. Only checkpoint-derived quantities enter the target-stage transfer procedure. The target sample is split once into $\mathcal{D}_0^{\mathrm{tr}}$ and $\mathcal{D}_0^{\mathrm{val}}$, of sizes n_{tr} and n_{val}. All model fitting and training-only tuning, including cross-fitting, use $\mathcal{D}_0^{\mathrm{tr}}$; $\mathcal{D}_0^{\mathrm{val}}$ is accessed only after the candidate predictors have been frozen. For classification, let C denote the number of target classes. The base checkpoint defines a representation $z_0(x) = z_{\theta_0}(x) \in \mathbb{R}^d$ at a prespecified fusion layer, followed by a target head $g_\beta : \mathbb{R}^d \to \mathbb{R}^C$; for regression, $g_\beta : \mathbb{R}^d \to \mathbb{R}$. Let $\ell(y, g_\beta(z))$ denote a differentiable prediction loss, such as cross-entropy for classification or squared error for regression. The fusion layer and source-block partition are fixed before target validation and therefore are not selected using $\mathcal{D}_0^{\mathrm{val}}$. A complete notation summary is provided in Appendix A.

For each source s, let $\mathcal{B}_s$ denote a prespecified partition into reusable architectural blocks, such as residual stages, adapters, LoRA modules, or groups of attention layers. Such blocks correspond to natural modular units in parameter-efficient adaptation and expert architectures (Shazeer et al., 2017; Houlsby et al., 2019; Dosovitskiy et al., 2021; Pfeiffer et al., 2021; Riquelme et al., 2021; Fedus et al., 2022; Hu et al., 2022). We index a source-block pair by $e = (s, b)$ and define $\mathcal{E} = \{(s, b) : s \in [S],\ b \in \mathcal{B}_s\}$, $M = |\mathcal{E}|$. Let $P_b\Delta_s$ retain the source update only on block b. The corresponding hybrid checkpoint is $\theta_e = \theta_0 + P_b\Delta_s$, and we define its *exact functional direction* at target input x by

$$d_e(x) = z_{\theta_e}(x) - z_0(x). \tag{2.1}$$

Thus $d_e(x)$ is the finite change at the designated fusion layer produced by inserting one source block into the base checkpoint; it is not a linearization of the network around θ_0. The decomposition is architectural but not spectral, so each direction remains tied to a prespecified source block. Any representation normalization is fixed in advance and applied identically to the base and hybrid representations. This prevents preprocessing differences from confounding directional comparisons. Equation (2.1) assumes that the base and source checkpoints share a parameterization, as in task-vector and PEFT settings. This common coordinate system makes each block substitution well defined and keeps functional comparisons on one fixed representation scale.

We next define the target direction against which each source-induced change is evaluated. Let $\beta^* \in \arg\min_\beta \mathbb{E}[\ell\{Y, g_\beta(z_0(X))\}]$ denote a population target head fitted to the base representation, and define the conditional target risk $r_x(z) = \mathbb{E}\{\ell(Y, g_{\beta^*}(z)) \mid X = x\}$. Differentiating this risk with respect to the representation gives the population descent direction

$$v^*(x) = -\nabla r_x\{z_0(x)\} = -\mathbb{E}[\nabla_z \ell\{Y, g_{\beta^*}(z_0(x))\} \mid X = x], \tag{2.2}$$

whenever differentiation and conditional expectation can be interchanged. Thus $v^*(x)$ is the representation-space direction of local target-risk descent. For multiclass cross-entropy with a linear-logit head $g_\beta(z) = Wz+b$, where $W \in \mathbb{R}^{C\times d}$ is the weight matrix and $b \in \mathbb{R}^C$ is the intercept vector, and with conditional class-probability vector $\eta(x) = (P_0(Y = 1 \mid X = x), \ldots, P_0(Y = C \mid X = x))^\top$, substituting the standard cross-entropy gradient into (2.2) gives

$$v^*(x) = W^\top\{\eta(x) - p_0(x)\} \text{ and } p_0(x) = \mathrm{softmax}\{Wz_0(x) + b\}. \tag{2.3}$$

Because $v^*(x)$ depends on the unknown conditional target law, we estimate it from target training data. We partition $\mathcal{D}_0^{\mathrm{tr}}$ into K_{cf} folds. For observation i in fold k, we fit the target head $\hat\beta^{(-k)}$ on the remaining folds and form the cross-fitted pseudo-gradient $G_i = -\nabla_z \ell\{Y_i, g_{\hat\beta^{(-k)}}(z_0(X_i))\}$. Let $v_\phi : \mathcal{X} \to \mathbb{R}^d$ be a direction-field model parameterized by ϕ, and let $\lambda_v \geq 0$ control its ℓ_2 regularization. We then estimate ϕ from the cross-fitted pseudo-gradients by

$$\hat\phi \in \arg\min_\phi \left\{ \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \|v_\phi(X_i) - G_i\|_2^2 + \lambda_v \|\phi\|_2^2 \right\}, \quad \hat v_\phi(x) = v_{\hat\phi}(x). \tag{2.4}$$

Cross-fitting ensures that an observation is not used to fit the head that generates its own pseudo-gradient (Chernozhukov et al., 2018); it serves here as a sample-splitting device but not as an invocation of orthogonal-score theory. For a fold-specific linear softmax head with weight matrix $\hat W^{(-k)}$, we have $G_i = \hat W^{(-k)\top}\{e_{Y_i} - \hat p_i^{(-k)}\}$, where e_{Y_i} is the one-hot class vector and $\hat p_i^{(-k)}$ is the fold-specific softmax probability vector at X_i. The field class and λ_v are selected using only inner splits of $\mathcal{D}_0^{\mathrm{tr}}$, and the prediction head is refit on all of $\mathcal{D}_0^{\mathrm{tr}}$ after the pseudo-gradients have been constructed.

The source-induced direction $d_e(x)$ and the estimated target descent field $\hat v_\phi(x)$ now provide the two ingredients needed to assess whether source block e is locally useful for input x. We use directional agreement to measure whether the block points towards target descent, and a curvature-corrected score to account additionally for the magnitude of the induced change. For a fixed stabilizer $\epsilon_{\mathrm{ang}} > 0$, define

$$\hat A_e(x) = \frac{\langle \hat v_\phi(x), d_e(x)\rangle}{(\|\hat v_\phi(x)\|_2 + \epsilon_{\mathrm{ang}})(\|d_e(x)\|_2 + \epsilon_{\mathrm{ang}})} \text{ and } \hat S_e(x) = \langle \hat v_\phi(x), d_e(x)\rangle - \frac{\hat L}{2}\|d_e(x)\|_2^2, \tag{2.5}$$

where $\hat L > 0$ is selected from a prespecified grid. The score $\hat A_e(x)$ measures local directional agreement while reducing sensitivity to nearly zero vectors, whereas $\hat S_e(x)$ also penalizes large induced changes and provides the curvature-aware score used to rank and screen candidate directions. The form of $\hat S_e(x)$ follows from the standard descent lemma for L-smooth functions. If r_x is L-smooth on the relevant representation neighborhood, then every admissible correction u satisfies

$$r_x\{z_0(x) + u\} - r_x\{z_0(x)\} \leq -\langle v^*(x), u\rangle + \frac{L}{2}\|u\|_2^2. \tag{2.6}$$

Hence positive alignment contributes first-order descent, while the quadratic term penalizes corrections whose magnitude is too large. Because both $d_e(x)$ and the target descent field vary with x, the same source block can have positive directional compatibility for some target inputs and negative compatibility for others. To clarify how this local functional geometry relates to conventional parameter-angle transfer, we next consider the corresponding linear model.

Proposition 2.1. *Suppose $f_\beta(x) = x^\top \beta$, squared loss is used, and a source update is Δ_s. If the target correction is Δ_0 and $\Sigma_X = \mathbb{E}(XX^\top)$, then the population functional alignment satisfies*

$$\mathbb{E}\{(X^\top \Delta_s)(X^\top \Delta_0)\} = \Delta_s^\top \Sigma_X \Delta_0.$$

When $\Sigma_X = I$ and $\Delta_s, \Delta_0 \neq 0$, normalized population functional alignment equals the ordinary cosine of the two parameter directions. More generally, if Σ_X is positive definite, the same normalization equals their cosine in the inner-product geometry induced by the target second-moment matrix Σ_X.

Proposition 2.1 connects local functional alignment to angle-based transfer. In the linear case, averaging functional effects over the target input distribution produces a single global geometry, and the ordinary parameter angle is recovered when that geometry is isotropic. In a deep network, by contrast, the finite functional change $d_e(x)$ depends nonlinearly on the input, so a single global parameter angle need not preserve the local directional information used by LTD.

3 Local Direction Selection and Sparse Composition

Section 2 defines how to evaluate the usefulness of a source block once its exact functional direction $d_e(x)$ is available. Evaluating every source block before making that decision, however, would require dense source-bank computation at every target input. LTD therefore separates *proposal* from *exact directional evaluation*. During target training, functional directions for all source blocks can be computed once on $\mathcal{D}_0^{\mathrm{tr}}$ and used to train a lightweight proposal scorer. At test time, the scorer first selects a small set of promising blocks using only the base representation and expert descriptors; exact functional directions are then evaluated only for those shortlisted blocks.

Associate each source block $e \in \mathcal{E}$ with an expert embedding $u_e \in \mathbb{R}^{d_u}$. Let d_u denote the embedding dimension, d_r the hidden width of the proposal network, and σ a fixed coordinatewise activation. The proposal parameters are $A_0 \in \mathbb{R}^{d_r \times d}$, $A_u \in \mathbb{R}^{d_r \times d_u}$, and $a, c \in \mathbb{R}^{d_r}$. For target input x, define the scalar proposal score $r_e(x)$ and its top-k shortlist by

$$r_e(x) = a^\top \sigma\left\{A_0 z_0(x) + A_u u_e + c\right\} \text{ and } J_k(x) = \mathrm{TopK}_{e \in \mathcal{E}}\, r_e(x), \tag{3.1}$$

where $k \leq M$ is the shortlist size and ties are resolved by a fixed deterministic rule. Thus $r_e(x)$ is inexpensive to evaluate because it does not require executing the hybrid checkpoint θ_e. The proposal scorer is trained to predict which source blocks are likely to have favorable local functional effects. For a candidate curvature value $\hat{L}$, define on each training pair (X_i, e) the cross-fitted corrected-gain target

$$\tilde{S}_{ie}^{(\hat{L})} = \langle G_i, d_e(X_i)\rangle - \frac{\hat{L}}{2}\,\|d_e(X_i)\|_2^2\,. \tag{3.2}$$

This is the training-sample analogue of the curvature-aware score in (2.5), with the cross-fitted pseudo-gradient G_i replacing the estimated conditional field. For each distinct $\hat{L}$ in the candidate grid, the proposal parameters, including the expert embeddings, are fitted with a pairwise ranking loss that assigns larger $r_e(X_i)$ to blocks with larger $\tilde{S}_{ie}^{(\hat{L})}$; Appendix B gives the pair construction and optimization details. All curvature values share the cached quantities G_i, $d_e(X_i)$, $\langle G_i, d_e(X_i)\rangle$, and $\|d_e(X_i)\|_2^2$. The proposal score is used only for computational screening. It is not itself treated as evidence that transfer is beneficial. For a new input x, after $J_k(x)$ is formed, LTD evaluates the exact $d_e(x)$ for $e \in J_k(x)$ and makes the transfer decision from those functional directions and the estimated target field $\hat{v}_\phi(x)$.

We write the ordered shortlist as $J_k(x) = (e_1, \ldots, e_k)$ and collect its exact functional directions in $D_k(x) = [d_{e_1}(x), \ldots, d_{e_k}(x)] \in \mathbb{R}^{d \times k}$. The coefficient vector $\alpha \in \mathbb{R}^k$ determines how the shortlisted directions are combined. We allow at most $q \leq k$ active directions and constrain their total weight by $A_{\max} > 0$. For ridge parameter $\lambda \geq 0$, LTD solves

$$\hat{\alpha}(x) \in \arg\min_{\alpha \geq 0,\ \|\alpha\|_0 \leq q,\ \|\alpha\|_1 \leq A_{\max}} \left\{ -\hat{v}_\phi(x)^\top D_k(x)\alpha + \frac{\hat{L}}{2}\,\|D_k(x)\alpha\|_2^2 + \frac{\lambda}{2}\,\|\alpha\|_2^2 \right\}. \tag{3.3}$$

The three terms have distinct roles. The first rewards a combined correction that aligns with estimated target descent; the second is the curvature penalty suggested by (2.6) and accounts jointly for the magnitudes and interactions of the selected functional directions; the third stabilizes the coefficients when shortlisted directions are redundant or nearly collinear. The constraint $\|\alpha\|_0 \leq q$ enforces sparse reuse,

while $\|\alpha\|_1 \le A_{\max}$ limits the aggregate correction. We restrict $\alpha \ge 0$ so that a source block can be attenuated or omitted but is not artificially sign-reversed; the composition reuses source-induced changes in the directions they actually produce. For $q \le 3$, the shortlisted bank is small enough to enumerate supports and solve the nonnegative ridge problem on each support, including the empty support. For larger q, we first retain the top-q candidates with positive $\hat{S}_e(x)$ and then solve the same joint nonnegative ridge problem on that support. Appendix B gives the solver details. The resulting correction is $D_k(x)\hat{\alpha}(x)$, and the transferred representation is

$$z_{\mathrm{LTD}}(x) = z_0(x) + \gamma D_k(x)\hat{\alpha}(x), \quad 0 \le \gamma \le \gamma_{\max} < \infty, \tag{3.4}$$

where γ is a transfer-amplitude parameter selected on the prespecified validation grid. The budget $A_{\max}$ constrains the sparse composition while it is being formed, whereas γ calibrates the strength of the correction after that composition rule has been specified. The zero vector is feasible in (3.3), and the amplitude grid contains $\gamma = 0$; consequently, target-only prediction is nested exactly.

For a configuration h, let $f_h(x) = g_{\hat{\beta}}\{z_{\mathrm{LTD},h}(x)\}$ denote its frozen predictor, and write $f_0(x) = g_{\hat{\beta}}\{z_0(x)\}$ for target-only prediction. Let Λ be a finite configuration grid fixed before $\mathcal{D}_0^{\mathrm{val}}$ is examined. Each $h \in \Lambda$ specifies $h = (k, q, A_{\max}, \lambda, \hat{L}, \gamma)$, together with the corresponding proposal scorer. The grid contains a $\gamma = 0$ configuration inducing f_0. Every candidate is trained using only $\mathcal{D}_0^{\mathrm{tr}}$ and is frozen before validation. Relabel the validation observations as (X_i, Y_i), $i = 1, \dots, n_{\mathrm{val}}$, and define the ordinary empirical validation risk

$$\hat{R}_{\mathrm{val}}(f_h) = \frac{1}{n_{\mathrm{val}}} \sum_{i=1}^{n_{\mathrm{val}}} \ell\{Y_i, f_h(X_i)\}. \tag{3.5}$$

Ordinary LTD selects the frozen configuration minimizing $\hat{R}_{\mathrm{val}}(f_h)$. Although validation selection targets improved average performance, a finite validation sample does not by itself rule out choosing a transfer configuration whose population risk exceeds that of target-only prediction. We therefore also consider an optional safety variant, LTD-SAFE, which certifies transfer against the target-only predictor using the untouched validation sample. Let $\ell_B \in [0, B]$ be a bounded loss fixed before validation and define its population target risk by $R_B(f) = \mathbb{E}_{P_0}[\ell_B\{Y, f(X)\}]$. For each candidate h, define its paired validation-loss difference from target-only by $Z_{ih} = \ell_B\{Y_i, f_h(X_i)\} - \ell_B\{Y_i, f_0(X_i)\}$, $\bar{Z}_h = n_{\mathrm{val}}^{-1} \sum_{i=1}^{n_{\mathrm{val}}} Z_{ih}$, and $\hat{s}_h^2 = (n_{\mathrm{val}} - 1)^{-1} \sum_{i=1}^{n_{\mathrm{val}}} (Z_{ih} - \bar{Z}_h)^2$. Thus $\bar{Z}_h < 0$ favors transfer on the bounded validation loss. For a prespecified failure probability $\delta \in (0, 1)$, let $L_\delta = \log\{2|\Lambda|/\delta\}$. The corresponding one-sided empirical-Bernstein upper bound is

$$U_h = \bar{Z}_h + \sqrt{\frac{2\hat{s}_h^2 L_\delta}{n_{\mathrm{val}}}} + \frac{14 B L_\delta}{3(n_{\mathrm{val}} - 1)}. \tag{3.6}$$

The quantity U_h is an upper confidence bound for the bounded-risk difference $R_B(f_h) - R_B(f_0)$, simultaneously over the prespecified finite candidate family. Hence a transfer candidate is certified only when $U_h \le 0$. Among certified candidates, LTD-SAFE selects the one with smallest $\hat{R}_{\mathrm{val}}(f_h)$; if none is certified, it returns f_0. Proposition 4.3 gives the corresponding finite-sample guarantee using the empirical-Bernstein inequality of Maurer and Pontil (2009) and a union bound.

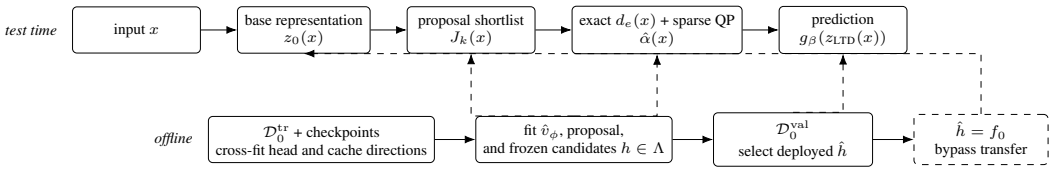


Figure 2: **Training, selection, and deployment of LTD.** Target training data fit the target head, direction field, and proposal scorers, while target validation selects one frozen deployment specification. For a test input, the selected proposal scorer shortlists source blocks, exact functional directions are evaluated only for that shortlist, and (3.3) forms the sparse correction. If the selected specification is f_0, the entire transfer path is bypassed.

The certification is global to the target task, not input-specific; it selects one frozen deployment specification before test inference begins. It controls only the prespecified bounded risk R_B and does not claim pointwise nonharmfulness or protection for every downstream metric. Because all candidate

predictors are frozen before validation and the confidence event is simultaneous over Λ, certified candidates may be ranked on the same validation sample without refitting. If the selected specification is target-only, proposal scoring and exact direction evaluation are bypassed at test time. Figure 2 summarizes how these components fit together, separating target-stage training and validation selection from the sparse test-time transfer computation.

Algorithm 1 Training, selection, and deployment of LTD

Require: Base checkpoint θ_0; source checkpoints $\{\theta_s\}_{s=1}^{S}$; fixed fusion layer and block partitions; target split $\mathcal{D}_0^{\mathrm{tr}}, \mathcal{D}_0^{\mathrm{val}}$; finite grid Λ; optional safety level δ.

1: Define the block-hybrid checkpoints $\theta_e = \theta_0 + P_b(\theta_s - \theta_0)$ for every $e = (s, b) \in \mathcal{E}$.
2: Partition $\mathcal{D}_0^{\mathrm{tr}}$ into K_{cf} folds; fit the fold-specific target heads and construct the cross-fitted pseudo-gradients G_i; then refit the prediction head on all of $\mathcal{D}_0^{\mathrm{tr}}$.
3: Using only inner splits of $\mathcal{D}_0^{\mathrm{tr}}$, select the direction-field class and λ_v, and fit $\hat{v}_\phi$ by (2.4).
4: For every training observation and source block, compute and cache $d_e(X_i)$, $\langle G_i, d_e(X_i)\rangle$, and $\|d_e(X_i)\|_2^2$.
5: For each distinct $\hat{L}$ appearing in Λ, form $\tilde{S}_{ie}^{(\hat{L})}$ by (3.2) and fit the corresponding proposal scorer in (3.1).
6: For every $h \in \Lambda$, freeze the resulting routed predictor: proposal shortlist $\to$ exact shortlisted directions $\to$ sparse composition (3.3) $\to$ amplitude scaling (3.4).
7: Evaluate all frozen candidates on $\mathcal{D}_0^{\mathrm{val}}$. Select the ordinary LTD minimizer of $\hat{R}_{\mathrm{val}}$, or apply (3.6) and use the LTD-SAFE certified selection rule.
8: **For a new input** x**:** if f_0 is selected, return $g_{\hat{\beta}}\{z_0(x)\}$. Otherwise compute $z_0(x)$, form $J_{\hat{k}}(x)$, evaluate only the shortlisted $d_e(x)$, solve (3.3) with the selected configuration, and return $g_{\hat{\beta}}\{z_{\mathrm{LTD}}(x)\}$.
9: **return** The frozen deployed predictor and its target-only fallback rule.

Algorithm 1 separates learning, selection, and deployment in LTD. Dense evaluation of the source bank is confined to the target training sample, where it provides supervision for the proposal scorer and the target direction field. After validation has fixed the deployment specification, a new input requires only the base forward pass, proposal scoring, exact evaluation of the shortlisted source blocks, and the sparse composition in (3.3). Thus the computational benefit comes from deciding which source computations merit exact evaluation before those computations are executed.

4 Theoretical Properties

We study LTD at the representation scale, treating the exact functional directions $d_e(x)$ as the transfer dictionary. This separates the geometric question, i.e., whether a feasible correction reduces target risk, from the optimization that produced the source checkpoints. Assume that r_x is differentiable and L-smooth on the relevant representation neighborhood. For the theoretical analysis, it is convenient to absorb the outer amplitude γ into the nonnegative direction coefficients. For sparsity level q and transfer budget $A_{\max}$, let $C_{\max} = \gamma_{\max} A_{\max}$ and define the population correction class

$$\mathcal{U}_q(x) = \left\{ \sum_{e \in \mathcal{E}} \omega_e d_e(x) : \omega \geq 0,\ \|\omega\|_0 \leq q,\ \|\omega\|_1 \leq C_{\max} \right\}. \tag{4.1}$$

Every correction produced by (3.4) belongs to this class for the corresponding q and $A_{\max}$, and $0 \in \mathcal{U}_q(x)$. By the smoothness inequality (2.6), the target-risk change of any $u \in \mathcal{U}_q(x)$ is bounded by

$$Q_x(u) = -\langle v^*(x), u\rangle + \frac{L}{2}\|u\|_2^2 \text{ and } u_q^*(x) \in \arg\min_{u \in \mathcal{U}_q(x)} Q_x(u). \tag{4.2}$$

Thus $u_q^*(x)$ is the ideal local correction under the quadratic smoothness bound. Define its population directional gain by $\Gamma_q = -\mathbb{E}_X Q_X\{u_q^*(X)\}$. Because the zero correction is feasible, $Q_x\{u_q^*(x)\} \leq Q_x(0) = 0$, and hence $\Gamma_q \geq 0$. To compare the learned correction with this oracle, we define the estimated quadratic objective

$$\hat{Q}_x(u) = -\langle \hat{v}_\phi(x), u\rangle + \frac{\hat{L}}{2}\|u\|_2^2. \tag{4.3}$$

It differs from Q_x through estimation of the target direction field and the smoothness constant. Theorem 4.1 quantifies how these discrepancies, together with proposal and optimization errors, affect the population risk of the deployed correction.

Theorem 4.1. *Condition on the fitted target-stage objects and fix the frozen deployed specification* $\hat{h} = (\hat{k}, \hat{q}, \hat{A}_{\max}, \hat{\lambda}, \hat{L}, \hat{\gamma})$. *In (4.1), we take* $q = \hat{q}$ *and* $C_{\max} = \gamma_{\max}\hat{A}_{\max}$. *Let* $\hat{u}(x) = \hat{\gamma} D_{\hat{k}}(x)\hat{\alpha}(x)$

be the deployed correction from (3.3)-(3.4), and assume that all oracle, approximating, and deployed corrections considered below have Euclidean norm at most R. Let $\tilde{u}_q(x)$ denote the best available approximation to $u_q^*(x)$ after proposal screening, and suppose $\mathbb{E}_X[Q_X\{\tilde{u}_q(X)\} - Q_X\{u_q^*(X)\}] \leq \varepsilon_{\mathrm{dict}}$. Suppose further that the deployed correction satisfies $\hat{Q}_x\{\hat{u}(x)\} \leq \hat{Q}_x\{\tilde{u}_q(x)\} + \epsilon_{\mathrm{alg}}(x)$, where $\epsilon_{\mathrm{alg}}(x) \geq 0$ accounts for ridge regularization, the finite amplitude grid, approximate support screening, and numerical optimization. Define $\varepsilon_v = \mathbb{E}_X \|\hat{v}_\phi(X) - v^*(X)\|_2$, $\varepsilon_L = R^2|\hat{L} - L|$, and $\varepsilon_{\mathrm{alg}} = \mathbb{E}_X \epsilon_{\mathrm{alg}}(X)$, where expectations are over a fresh target draw $X \sim P_0$, conditional on the fitted objects. Then

$$\mathbb{E}_X[r_X\{z_0(X) + \hat{u}(X)\} - r_X\{z_0(X)\}] \leq -\Gamma_q + 2R\varepsilon_v + \varepsilon_{\mathrm{dict}} + \varepsilon_L + \varepsilon_{\mathrm{alg}}. \tag{4.4}$$

Theorem 4.1 decomposes the gap from the local directional oracle into four sources. The term ε_v measures estimation of the target descent field, $\varepsilon_{\mathrm{dict}}$ records oracle directions lost through proposal screening or unavailable in the implemented dictionary, ε_L measures error in the selected smoothness surrogate, and $\varepsilon_{\mathrm{alg}}$ collects the remaining restrictions of the implemented composition rule. Consequently, if Γ_q exceeds their sum, the learned correction strictly reduces population target risk relative to z_0 under the population head g_{β^*}. If $\Gamma_q = 0$, target-only is an oracle-optimal feasible choice, but (4.4) alone does not certify that an estimated correction is nonharmful. The representation-scale formulation does not assume that parameter addition is functionally linear; each $d_e(x)$ in (2.1) is the exact finite functional change produced by the corresponding hybrid block. The selection of $\hat{L}$ and the finite implementation restrictions enter explicitly through ε_L and $\varepsilon_{\mathrm{alg}}$, with neither term hidden in an oracle assumption. Appendix A gives the proof and a statistical specialization for ε_v.

The above bound quantifies how closely LTD approaches a local oracle, but it does not yet establish that locality can be necessary. The next result isolates this distinction by exhibiting a problem in which every input-independent combination necessarily averages away useful region-specific information.

Theorem 4.2. *Let the target input belong to two regions G_1 and G_2, each with probability $1/2$, and consider scalar corrections with conditional squared risk $r_x(u) = \frac{1}{2}(1-u)^2$. Suppose the two source directions satisfy $d_1(G_1) = 1,\ d_1(G_2) = -1,\ d_2(G_1) = -1,$ and $d_2(G_2) = 1$. For any input-independent coefficients $a_1, a_2 \in \mathbb{R}$, the correction $a_1 d_1(X) + a_2 d_2(X)$ has population risk $\frac{1}{2}\left\{1 + (a_1 - a_2)^2\right\} \geq \frac{1}{2}$. In contrast, the local rule that selects d_1 on G_1 and d_2 on G_2 has population risk zero. Thus every input-independent source-weight rule has excess squared risk at least $1/2$ relative to the local directional oracle, even though both sources have zero global average directional alignment with the desired correction. A positive constant separation persists under sufficiently small bounded perturbations over any fixed compact set of input-independent coefficients.*

The separation is structural and not an estimation artifact. Even exact knowledge of each global source alignment cannot reveal the opposing local effects removed by averaging. Improving estimation of a single global merge weight therefore cannot eliminate this cancellation; the missing information is the dependence of transferability on the target input. Together with the directional-gain bound, this result explains both when local transfer can help and why a global rule can fail. Neither result, however, provides a finite-sample nonharmfulness guarantee for the learned predictor. The optional validation gate in Section 3 provides that separate protection.

Proposition 4.3. *Condition on the finite frozen candidate family $\{f_h : h \in \Lambda\}$, constructed without using $\mathcal{D}_0^{\mathrm{val}}$. Suppose $n_{\mathrm{val}} \geq 2$, the validation observations are independent and identically distributed from P_0, $0 \leq \ell_B \leq B$, and $\delta \in (0,1)$. With U_h defined in (3.6), with probability at least $1 - \delta$, simultaneously for every $h \in \Lambda$,*

$$R_B(f_h) - R_B(f_0) \leq U_h.$$

Consequently, writing $\hat{f}_{\mathrm{safe}}$ for the predictor returned by LTD-SAFE*, on the same event $R_B(\hat{f}_{\mathrm{safe}}) \leq R_B(f_0)$. If no transfer candidate satisfies $U_h \leq 0$, the deployed predictor is exactly f_0.*

Proposition 4.3 concerns only the prespecified bounded loss R_B; it does not imply pointwise nonharmfulness or protection for every downstream metric. The result requires neither source-target similarity nor correct specification of the direction field or proposal mechanism. The tradeoff is conservativeness: when n_{val} is small, the confidence bound can remain positive even for a beneficial transfer candidate, causing LTD-SAFE to fall back to target-only prediction. This conservativeness is intentional, since the gate is designed to prevent uncertain transfer but not to maximize the frequency with which transfer is accepted. Certification prioritizes bounded-risk control over acceptance.

5 NUMERICAL STUDIES

We evaluate LTD in two complementary settings. A controlled simulation first isolates the proposed local geometry when the useful source directions are known. We then evaluate pretrained vision and language models across architectures, tasks, and domains, where functional directions arise from source checkpoints and source examples are unavailable during target adaptation. Complete designs, tuning grids, additional diagnostics, and per-dataset results are given in the appendix.

5.1 MONTE CARLO SIMULATIONS

We consider a four-class problem with $X \in \mathbb{R}^{12}$ and four equally likely regions determined by the signs of (X_1, X_2). The source bank contains $M = 12$ functional directions; two are useful in each region but harmful outside their home region, and four are distractors. Thus local transfer is informative while global averaging cancels opposing effects. This construction isolates directional cancellation. It permits direct evaluation of local support recovery. The target direction field is estimated by multinomial logistic regression using target training data, and local methods use $q = 2$. Unless varied, $n_{\mathrm{tr}} = 400$, $n_{\mathrm{val}} = 300$, and $n_{\mathrm{test}} = 2000$. Table 1 averages 500 independent replications.

Table 1: **Partial-transfer experiment.** Log loss and accuracy are test metrics; Route F1 measures agreement between the learned pre-gate route and the two region-matched source directions. Fallback is the LTD-SAFE target-only fallback frequency. Standard errors are in parentheses.

Method	Log loss↓	Acc.↑	Route F1↑	Fallback↓
Target only	1.261(.001)	.424(.001)	–	–
Dense transfer	1.261(.001)	.424(.001)	–	–
Global angle	1.261(.001)	.424(.001)	–	–
Local cosine	1.113(.002)	.525(.001)	.879	–
LTD	1.113(.002)	.524(.001)	.879	–
LTD-SAFE	1.133(.003)	.517(.002)	.879	.02
Oracle target field	1.080(.002)	.542(.001)	.995	–

Dense transfer and the global-angle rule select $\gamma = 0$ on validation and coincide with target-only prediction, whereas local alignment reduces log loss from 1.261 to 1.113 and raises accuracy from $.424$ to about $.525$. Local cosine and LTD perform nearly identically, indicating that the principal gain in this transparent design comes from recovering the correct input-specific directions without requiring fine adjustment of their weights. This pattern directly isolates locality as the operative transfer mechanism. LTD-SAFE falls back in only 2% of replications and gives up a modest amount of average improvement in exchange for certification of the prespecified clipped loss.

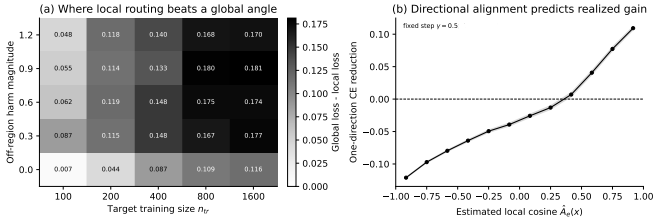


Figure 3: **Mechanism diagnostics.** (a) Mean test-log-loss advantage of local over global directional transfer as off-region harm and target sample size vary; each cell averages 500 replications. (b) Realized one-direction cross-entropy reduction versus estimated local alignment at fixed $\gamma = 0.5$; the band is a 95% normal interval for the binned mean. Stronger off-region harm increases the value of locality, while larger estimated alignment predicts larger realized gain.

Figure 3 further tests the mechanism. The local advantage also strengthens systematically with the amount of target information. When n_{tr} increases from 100 to 1600, Route F1 for the local rule rises from .654 to .981, while its mean test log loss falls from 1.189 to 1.081; the global-angle baseline remains essentially unchanged near the target-only risk. The behavior of LTD-SAFE is complementary. At $n_{\mathrm{tr}} = 100$, uncertainty in the learned target field leads to fallback in 55% of replications, whereas this frequency falls to 5% at $n_{\mathrm{tr}} = 200$ and essentially vanishes once the local directions are estimated more accurately. Thus additional target information improves not only prediction but also recovery of the local routing structure and the ability of the validation gate to certify transfer. These patterns are consistent with Theorem 4.1; as direction-field error decreases, the useful local signal becomes increasingly dominant. Falling fallback rates also corroborate increasingly reliable local transfer decisions.

5.2 PRETRAINED-MODEL EVALUATION

We next evaluate whether the mechanism persists when directions are induced by pretrained checkpoints. CLIP (Radford et al., 2021; Ilharco et al., 2023) treats Cars, DTD, EuroSAT, GTSRB, MNIST, RESISC45, SVHN, and SUN397 in turn as low-resource targets, with the remaining checkpoints forming the source bank. PACS and OfficeHome (Li et al., 2017; Venkateswara et al., 2017) provide held-out-domain experiments with shared ResNet/ViT backbones, while a T5/RoBERTa suite (Liu et al., 2019; Raffel et al., 2020) evaluates source LoRA/adapters across language tasks. Across families, the transfer unit is a reusable source computation whose effect is evaluated in the target representation space, without reliance on any source-level score. Functional directions are always measured in a shared representation before the target-specific head, and methods receive the same target-label budget. Headline results use the prespecified middle budget in each family: 64 labels per class for CLIP, 50% of the fixed low-resource pool for PACS and OfficeHome, and 128 labeled examples in total for language tasks. Comparisons include target-only adaptation, best-source scoring, task arithmetic, DARE, TIES-Merging, AdaMerging (Yadav et al., 2023; Yang et al., 2024b; Yu et al., 2024), global parameter and functional angles, a parameter-matched local router, LTD, LTD-SAFE, and a full-bank LTD diagnostic without proposal screening. The primary metric is the prespecified benchmark metric; top-1 accuracy is used for image classification. We additionally report log loss, negative-transfer frequency (NT), exact directions evaluated per input, and relative inference FLOPs. Appendix E gives the frozen data-access, tuning, and compute-accounting rules.

Table 2: **Pretrained-model comparison.** NT is the fraction of target-task/seed pairs worse than paired target-only fits. "Eval. dirs." is the mean exact functional directions executed per input. Full-bank LTD uses the same estimated target field, sparse-composition rule, target data, and validation protocol as LTD, but skips proposal screening and evaluates all M directions before composition.

Method	Metric↑	Loss↓	NT↓	Eval. dirs.↓	FLOPs↓
Target only	80.9	.615	–	0	1.00
Best source	79.8	.642	.322	0	1.00
Task arithmetic	80.6	.602	.244	0	1.00
DARE	81.0	.591	.211	0	1.00
TIES-Merging	81.5	.579	.183	0	1.00
AdaMerging	82.1	.557	.139	0	1.00
Global parameter angle	81.7	.570	.167	0	1.00
Global functional angle	82.4	.544	.111	0	1.00
Parameter-matched local router	82.7	.532	.097	2.6	1.70
LTD	83.9	.487	.056	2.7	1.72
LTD-SAFE	83.7	.493	.019	2.3	1.65
Full-bank LTD (diagnostic)	84.1	.482	.049	all M	7.80

At the aggregate benchmark level, Table 2 shows a benefit from parameter geometry to functional geometry and then input-conditional transfer. Global functional alignment improves the metric from 81.7 to 82.4 and reduces NT from $.167$ to $.111$. Adding local routing and directional supervision raises the metric to 83.9, with log loss $.487$ and NT $.056$, showing that the gain is not explained by routing alone but by conditioning transfer on functional effects relative to target descent. Exhaustive evaluation is unnecessary; full-bank LTD improves the metric only to 84.1 and log loss to $.482$, while increasing FLOPs from 1.72 to 7.80. The small predictive gain from full-bank evaluation, despite more than a fourfold increase in compute, indicates that proposal screening preserves most of the useful source-bank information. LTD-SAFE further reduces NT to $.019$ with only a $.2$-point decrease in the primary metric and lower inference cost through target-only fallback. These results illustrate the statistical-computational tradeoff between proposal screening, local functional alignment, and safe deployment.

6 CONCLUDING REMARKS

LTD treats transferability as a local relation between a reusable source computation's functional change and the target loss-descent direction at a given input. This leads to proposal before exact evaluation, sparse local composition, and an optional target-only safety gate. The theory characterizes approximation to a local directional oracle, the limits of input-independent source weights, and finite-sample fallback protection, while experiments show that local functional alignment can recover useful computations hidden by global averaging using only part of the source bank. These results support organizing transfer around transferable computations, not entire source models, while richer functional dictionaries and heterogeneous source architectures remain for future work.

AI USE STATEMENT

Generative AI was used solely for assistance with manuscript wording and language refinement. It was not used in developing the research methods, theoretical arguments, or numerical findings. All text produced with AI assistance was carefully checked and edited by the authors for correctness, clarity, and proper attribution. The authors remain fully responsible for the final presentation of the manuscript.

REPRODUCIBILITY STATEMENT

Appendix provides the assumptions and proofs, algorithmic and implementation details, controlled data-generating process, fixed benchmark protocol, tuning grids, and additional diagnostic procedures, including sensitivity analyses and source-corruption checks reported therein. To preserve anonymity during double-blind review, all code used to reproduce the numerical results will be released publicly upon completion of the peer-review process.

ACKNOWLEDGMENTS

The authors gratefully acknowledge support from XXX and thank seminar and conference participants at XXX for helpful comments and discussions.

REFERENCES

Achille, A., Lam, M., Tewari, R., Ravichandran, A., Maji, S., Fowlkes, C., Soatto, S., and Perona, P. (2019). Task2Vec: Task embedding for meta-learning. *IEEE/CVF International Conference on Computer Vision*, 6429-6438.

Chen, S., Jie, Z., and Ma, L. (2024). LLaVA-MoLE: Sparse mixture of LoRA experts for mitigating data conflicts in instruction finetuning MLLMs. *arXiv:2401.16160*.

Chen, X., Wang, S., Fu, B., Long, M., and Wang, J. (2019). Catastrophic forgetting meets negative transfer: Batch spectral shrinkage for safe transfer learning. *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 1908-1918.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., and Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1), C1-C68.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations*.

Fedus, W., Zoph, B., and Shazeer, N. (2022). Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120), 1-39.

Feng, W., Hao, C., Zhang, Y., Han, Y., and Wang, H. (2024). Mixture-of-LoRAs: An efficient multitask tuning for large language models. *arXiv:2403.03432*.

Ferbach, D., Goujaud, B., Gidel, G., and Dieuleveut, A. (2024). Proving linear mode connectivity of neural networks via optimal transport. *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, 238, 3853-3861.

Frankle, J., Dziugaite, G. K., Roy, D., and Carbin, M. (2020). Linear mode connectivity and the lottery ticket hypothesis. *Proceedings of the 37th International Conference on Machine Learning*, 119, 3259-3269.

Gu, T., Han, Y., and Duan, R. (2025). Robust angle-based transfer learning in high dimensions. *Journal of the Royal Statistical Society Series B*, 87(3), 723-745.

He, Y., Hu, Y., Lin, Y., Zhang, T., and Zhao, H. (2024). Localize-and-Stitch: Efficient model merging via sparse task arithmetic. *Transactions on Machine Learning Research*.

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., and Gelly, S. (2019). Parameter-efficient transfer learning for NLP. *Proceedings of the 36th International Conference on Machine Learning*, 97, 2790-2799.

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

Huang, L.-K., Huang, J., Rong, Y., Yang, Q., and Wei, Y. (2022). Frustratingly easy transferability estimation. *Proceedings of the 39th International Conference on Machine Learning*, 162, 9201-9225.

Ilharco, G., Ribeiro, M. T., Wortsman, M., Schmidt, L., Hajishirzi, H., and Farhadi, A. (2023). Editing models with task arithmetic. *International Conference on Learning Representations*.

Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. (2019). Similarity of neural network representations revisited. *Proceedings of the 36th International Conference on Machine Learning*, 97, 3519-3529.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. (2017). Deeper, broader and artier domain generalization. *IEEE International Conference on Computer Vision*, 5543-5551.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv:1907.11692*.

Lu, Z., Fan, C., Wei, W., Qu, X., Chen, D., and Cheng, Y. (2024). Twin-Merging: Dynamic integration of modular expertise in model merging. *Advances in Neural Information Processing Systems*, 37.

Matena, M. S. and Raffel, C. (2022). Merging models with Fisher-weighted averaging. *Advances in Neural Information Processing Systems*, 35.

Maurer, A. and Pontil, M. (2009). Empirical Bernstein bounds and sample variance penalization. *The 22nd Conference on Learning Theory*.

Muqeeth, M., Liu, H., and Raffel, C. (2024). Soft merging of experts with adaptive routing. *Transactions on Machine Learning Research*.

Nguyen, C., Hassner, T., Seeger, M., and Archambeau, C. (2020). LEEP: A new measure to evaluate transferability of learned representations. *Proceedings of the 37th International Conference on Machine Learning*, 119, 7294-7305.

Pfeiffer, J., Kamath, A., Rücklé, A., Cho, K., and Gurevych, I. (2021). AdapterFusion: Non-destructive task composition for transfer learning. *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, 487-503.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning transferable visual models from natural language supervision. *Proceedings of the 38th International Conference on Machine Learning*, 139, 8748-8763.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21, 1-67.

Riquelme, C., Puigcerver, J., Mustafa, B., Neumann, M., Jenatton, R., Susano Pinto, A., Keysers, D., and Houlsby, N. (2021). Scaling vision with sparse mixture of experts. *Proceedings of the 35th International Conference on Neural Information Processing Systems*, 8583-8595.

Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., and Dean, J. (2017). Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *International Conference on Learning Representations*.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. (2017). Deep hashing network for unsupervised domain adaptation. *IEEE Conference on Computer Vision and Pattern Recognition*, 5385-5394.

Wang, H., Polo, F. M., Sun, Y., Kundu, S., Xing, E. P., and Yurochkin, M. (2024). Fusing models with complementary expertise. *International Conference on Learning Representations*.

Wang, K., Dimitriadis, N., Ortiz-Jiménez, G., Fleuret, F., and Frossard, P. (2024). Localizing task information for improved model merging and compression. *Proceedings of the 41st International Conference on Machine Learning* , 235, 50268-50287.

Wang, Z., Dai, Z., Poczos, B., and Carbonell, J. (2019). Characterizing and avoiding negative transfer. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11293-11302.

Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., and Schmidt, L. (2022). Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. *Proceedings of the 39th International Conference on Machine Learning*, 162, 23965-23998.

Yadav, P., Tam, D., Choshen, L., Raffel, C., and Bansal, M. (2023). TIES-Merging: Resolving interference when merging models. *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 7093-7115.

Yan, K., Zhang, M., Cui, S., Qu, Z., Jiang, B., Liu, F., and Zhang, C. (2025). CALM: Consensus-aware localized merging for multi-task learning. *Proceedings of the 42nd International Conference on Machine Learning*, 267, 70309-70329.

Yang, E., Shen, L., Wang, Z., Guo, G., Chen, X., Wang, X., and Tao, D. (2024a). Representation surgery for multi-task model merging. *Proceedings of the 41st International Conference on Machine Learning*, 235, 56332-56356.

Yang, E., Wang, Z., Shen, L., Liu, S., Guo, G., Wang, X., and Tao, D. (2024b). AdaMerging: Adaptive model merging for multi-task learning. *International Conference on Learning Representations*.

You, K., Liu, Y., Wang, J., and Long, M. (2021). LogME: Practical assessment of pre-trained models for transfer learning. *Proceedings of the 38th International Conference on Machine Learning*, 139, 12133-12143.

Yu, L., Yu, B., Yu, H., Huang, F., and Li, Y. (2024). Language models are super mario: Absorbing abilities from homologous models as a free lunch. *Proceedings of the 41st International Conference on Machine Learning*, 57755-57775.

APPENDIX

A THEORETICAL DETAILS AND PROOFS

This part gives the formal details underlying the results in Sections 2-4. We first state the probability convention, oracle classes, and regularity conditions used by the directional analysis. We then prove the directional-gain bound, the local-global separation result, and the finite-sample safety guarantee, and give a statistical specialization for estimation of the target direction field. Finally, we prove the linear reduction in Proposition 2.1 and clarify the scope of the representation-scale guarantees. Throughout the appendix, the base checkpoint θ_0 and source checkpoints $\{\theta_s : s \in [S]\}$ are treated as fixed after source training. Equivalently, all probability statements may be read conditionally on these checkpoints. Source examples therefore play no role in the target-stage probability space. The target observations are sampled from P_0, and the training and validation portions of the target sample are used exactly as specified in Sections 2 and 3.

A.1 FORMAL CLASSES, NOTATION, AND REGULARITY CONDITIONS

Table A.1 collects the notation used repeatedly in the main text and the proofs below. Symbols appearing only locally in a proof are defined where needed.

Table A.1: **Notation Summary**

Symbol	Meaning
$P_0, \mathcal{D}_0^{\mathrm{tr}}, \mathcal{D}_0^{\mathrm{val}}$	target distribution, target training split, and untouched target validation split
$n_0, n_{\mathrm{tr}}, n_{\mathrm{val}}$	total, training, and validation target sample sizes
S, C, d	numbers of source checkpoints and target classes, and fusion-layer representation dimension
$\theta_0, \theta_s, \Delta_s$	base checkpoint, source checkpoint, and source update $\Delta_s = \theta_s - \theta_0$
$\mathcal{B}_s, e = (s, b), \mathcal{E}, M$	prespecified source blocks, source-block index, complete source-block bank, and bank size $M = \lvert\mathcal{E}\rvert$
P_b, θ_e	block coordinate projection and hybrid checkpoint $\theta_e = \theta_0 + P_b\Delta_s$
$z_\theta(x), z_0(x), d_e(x)$	fusion representation, base representation, and exact finite functional direction $d_e(x) = z_{\theta_e}(x) - z_0(x)$
$g_\beta, \beta^*, \ell, r_x$	target prediction head, population target-head parameter, task loss, and conditional target risk
$W, b, \eta(x), p_0(x)$	linear-softmax weight matrix, intercept, target conditional class probabilities, and base-head class probabilities
$v^*(x), G_i, \hat{v}_\phi(x)$	population target descent field, cross-fitted pseudo-gradient, and estimated target direction field
$K_{\mathrm{cf}}, \lambda_v$	number of cross-fitting folds and direction-field ℓ_2 regularization weight
$\epsilon_{\mathrm{ang}}, \hat{A}_e(x), \hat{S}_e(x)$	angular stabilizer, stabilized directional-alignment score, and curvature-aware source score
$\tilde{S}_{ie}^{(L)}$	cross-fitted proposal-ranking target for observation i and source block e
$u_e, d_u, d_r, r_e(x), J_k(x)$	expert embedding, embedding dimension, proposal hidden width, proposal score, and top-k shortlist
$D_k(x), \alpha(x), k, q, A_{\max}$	shortlisted direction matrix, composition coefficients, shortlist size, sparsity budget, and ℓ_1 composition budget
$\lambda, \hat{L}, \gamma, \gamma_{\max}$	composition ridge parameter, selected curvature surrogate, transfer amplitude, and maximum transfer amplitude
Λ, h, f_h, f_0	finite validation grid, a frozen candidate specification, its predictor, and the target-only predictor
ℓ_B, B, R_B	prespecified bounded certification loss, its upper bound, and its population risk
$Z_{ih}, \bar{Z}_h, \hat{s}_h^2, L_\delta, U_h$	paired validation contrast, empirical mean and variance, multiplicity factor, and simultaneous empirical-Bernstein upper bound
$\delta, \hat{f}_{\mathrm{safe}}$	safety failure probability and predictor returned by LTD-SAFE
$C_{\max}, \mathcal{U}_q(x)$	population transfer budget $C_{\max} = \gamma_{\max}A_{\max}$ and corresponding sparse correction class
$Q_x(u), u_q^*(x), \Gamma_q$	population quadratic upper-bound objective, local directional oracle, and population directional gain
$\hat{Q}_x(u), \tilde{u}_q(x), \hat{u}(x)$	estimated quadratic objective, best available correction after proposal screening, and deployed correction
R, B_d	uniform bounds for feasible corrections and individual functional directions
$\varepsilon_v, \varepsilon_{\mathrm{dict}}, \varepsilon_L, \varepsilon_{\mathrm{alg}}$	direction-field, dictionary/proposal, smoothness-surrogate, and implementation errors in Theorem 4.1
Σ_X	target second-moment matrix $\mathbb{E}(XX^\top)$ in Proposition 2.1

The distinction between the source-block direction and the eventual correction is useful throughout the analysis. For $e = (s, b)$, $d_e(x) = z_{\theta_0+P_b\Delta_s}(x) - z_0(x)$ is the exact finite representation change generated by inserting source block e. No Taylor or neural-tangent approximation is used. By contrast, a correction is a nonnegative sparse combination of such directions. For theoretical purposes, as in Section 4, the outer transfer amplitude is absorbed into the direction coefficients. Let

$$C_{\max} = \gamma_{\max}A_{\max} \text{ and } \mathcal{U}_q(x) = \left\{ \sum_{e\in\mathcal{E}} \omega_e d_e(x) : \omega \geq 0,\ \|\omega\|_0 \leq q,\ \|\omega\|_1 \leq C_{\max} \right\}. \quad \text{(A.1)}$$

The zero correction belongs to $\mathcal{U}_q(x)$. Moreover, every correction produced by (3.4) belongs to the corresponding class (A.1), since $\omega_e = \gamma\alpha_e$ gives $\|\omega\|_1 \le \gamma_{\max} A_{\max}$. We impose only the regularity needed for the representation-scale directional argument.

Assumption 1. *The target observations are independent draws from P_0, and the indices forming $\mathcal{D}_0^{\mathrm{tr}}$ and $\mathcal{D}_0^{\mathrm{val}}$ are fixed independently of their observed values. The maps $x \mapsto z_0(x)$, $x \mapsto d_e(x)$, and $x \mapsto v^*(x)$ are measurable for every $e \in \mathcal{E}$. Moreover, $\mathbb{E}_X \|v^*(X)\|_2 < \infty$. The source bank $\mathcal{E}$, fusion layer, and source-block partition are fixed before target validation.*

Assumption 1 is a probability-space convention and imposes no source-target similarity assumption. In particular, it places no restriction on how close a source checkpoint must be to the target task. Treating the source checkpoints as fixed is natural here because the target-stage method has access to the checkpoints but not to the source examples that produced them.

Assumption 2. *For P_0-almost every x, the conditional risk r_x is differentiable on an open neighborhood $\mathcal{N}_x$ containing $\{z_0(x) + tu : u \in \mathcal{U}_q(x),\ 0 \le t \le 1\}$, and its gradient is L-Lipschitz there*

$$\|\nabla r_x(z) - \nabla r_x(z')\|_2 \le L \|z - z'\|_2, \quad z, z' \in \mathcal{N}_x,$$

with the same finite constant L for P_0-almost every x.

The requirement is local. It is needed only along line segments joining the base representation to feasible transferred representations. It does not assume global convexity of r_x, global smoothness of the network, or that the source checkpoints lie in a common convex basin. The analysis is therefore compatible with the exact nonlinear functional changes $d_e(x)$ used by LTD.

Assumption 3. *For some $B_d < \infty$, $\sup_{e\in\mathcal{E}} \|d_e(x)\|_2 \le B_d$ for P_0-almost every x. All oracle, approximating, and deployed corrections in Theorem 4.1 have Euclidean norm at most $R < \infty$.*

For the exact functional dictionary in (2.1), the second part follows from the first with the explicit choice $R = C_{\max} B_d$, because for every $u \in \mathcal{U}_q(x)$,

$$\|u\|_2 \le \sum_{e\in\mathcal{E}} \omega_e \|d_e(x)\|_2 \le C_{\max} B_d.$$

We state the correction bound separately to distinguish bounded source directions from the uniform correction radius used throughout the perturbation analysis. The next result establishes that the local oracle in Section 4 is well-defined.

Lemma A.1. *Under Assumptions 1 and 3, for P_0-almost every x, the set $\mathcal{U}_q(x)$ is nonempty and compact. Hence the continuous function*

$$Q_x(u) = -\langle v^*(x), u\rangle + \frac{L}{2}\|u\|_2^2$$

attains its minimum on $\mathcal{U}_q(x)$. A measurable minimizer $u_q^*(x) \in \arg\min_{u\in\mathcal{U}_q(x)} Q_x(u)$ can be chosen. Moreover,

$$Q_x\{u_q^*(x)\} \le 0 \text{ and } \Gamma_q = -\mathbb{E}_X Q_X\{u_q^*(X)\} \ge 0.$$

Proof. Define the coefficient set

$$\Omega_q = \left\{\omega \in \mathbb{R}^M : \omega \ge 0,\ \|\omega\|_0 \le q,\ \|\omega\|_1 \le C_{\max}\right\}.$$

Because $M < \infty$, Ω_q is the finite union, over supports $J \subseteq \mathcal{E}$ with $|J| \le q$, of the compact simplices

$$\left\{\omega \in \mathbb{R}^M : \omega_e = 0 \text{ for } e \notin J,\ \omega_e \ge 0,\ \sum_{e\in J}\omega_e \le C_{\max}\right\}.$$

Hence Ω_q is nonempty and compact. For (x, ω), define

$$u(x,\omega) = \sum_{e\in\mathcal{E}} \omega_e d_e(x) \text{ and } F(x,\omega) = Q_x\{u(x,\omega)\}.$$

For every fixed x, the map $\omega \mapsto u(x,\omega)$ is linear and continuous, and therefore $\omega \mapsto F(x,\omega)$ is continuous. For every fixed ω, measurability of $v^*(x)$ and $\{d_e(x) : e \in \mathcal{E}\}$ implies that $x \mapsto F(x,\omega)$

is measurable. Thus F is a Carathéodory function on the measurable input space $\mathcal{X}$ and the fixed compact set Ω_q. Moreover, $\mathcal{U}_q(x) = \{u(x,\omega) : \omega \in \Omega_q\}$. Consequently, compactness of Ω_q and continuity of $u(x,\cdot)$ imply that $\mathcal{U}_q(x)$ is nonempty and compact. Continuity of Q_x then implies that the minimum of Q_x over $\mathcal{U}_q(x)$ is attained.

By the measurable maximum theorem, the correspondence $x \mapsto \arg\min_{\omega\in\Omega_q} F(x,\omega)$ admits a measurable selector $\omega_q^*(x)$. Define $u_q^*(x) = \sum_{e\in\mathcal{E}} \omega_{q,e}^*(x) d_e(x)$. Then $u_q^*(x)$ is measurable and satisfies

$$u_q^*(x) \in \arg\min_{u\in\mathcal{U}_q(x)} Q_x(u)$$

for P_0-almost every x. Finally, $\omega = 0$ belongs to Ω_q, so $0 \in \mathcal{U}_q(x)$ and therefore

$$Q_x\{u_q^*(x)\} \le Q_x(0) = 0.$$

Under the stated integrability condition, $Q_X\{u_q^*(X)\}$ is integrable, and hence

$$\Gamma_q = -\mathbb{E}_X Q_X\{u_q^*(X)\} \ge 0.$$

□

Lemma A.1 ensures that the population local oracle is well defined. We next connect its quadratic objective to the actual change in conditional target risk. The following standard consequence of Assumption 2 is the basic inequality underlying both the oracle definition and the directional-gain analysis.

Lemma A.2. *Under Assumption 2, for P_0-almost every x and every $u \in \mathcal{U}_q(x)$,*

$$r_x\{z_0(x)+u\} - r_x\{z_0(x)\} \le -\langle v^*(x), u\rangle + \frac{L}{2}\|u\|_2^2 = Q_x(u).$$

Proof. We fix x in the P_0-almost-sure set of Assumption 2 and let $u \in \mathcal{U}_q(x)$. By that assumption, the line segment $\{z_0(x) + tu : 0 \le t \le 1\}$ lies in a neighborhood on which ∇r_x is L-Lipschitz. The descent lemma therefore gives

$$r_x\{z_0(x)+u\} \le r_x\{z_0(x)\} + \langle \nabla r_x\{z_0(x)\}, u\rangle + \frac{L}{2}\|u\|_2^2.$$

Since $v^*(x) = -\nabla r_x\{z_0(x)\}$, we obtain

$$r_x\{z_0(x)+u\} - r_x\{z_0(x)\} \le -\langle v^*(x), u\rangle + \frac{L}{2}\|u\|_2^2 = Q_x(u).$$

□

The inequality is one-sided; it provides a sufficient upper bound on the risk change induced by a feasible correction, but not an exact representation of that change. The next step is therefore to quantify how this quadratic upper bound is perturbed when the unknown target descent field and smoothness constant are replaced by their fitted counterparts.

Lemma A.3. *Let $\hat{Q}_x(u) = -\langle \hat{v}_\phi(x), u\rangle + \frac{\hat{L}}{2}\|u\|_2^2$. For every u satisfying $\|u\|_2 \le R$,*

$$\left|\hat{Q}_x(u) - Q_x(u)\right| \le R\,\|\hat{v}_\phi(x) - v^*(x)\|_2 + \frac{R^2}{2}|\hat{L} - L|.$$

Proof. Direct subtraction gives

$$\hat{Q}_x(u) - Q_x(u) = -\langle \hat{v}_\phi(x) - v^*(x), u\rangle + \frac{\hat{L}-L}{2}\|u\|_2^2.$$

Hence, by the triangle inequality and Cauchy-Schwarz,

$$\left|\hat{Q}_x(u) - Q_x(u)\right| \le |\langle \hat{v}_\phi(x) - v^*(x), u\rangle| + \frac{1}{2}|\hat{L}-L|\,\|u\|_2^2$$

and therefore

$$\left|\hat{Q}_x(u) - Q_x(u)\right| \le R\,\|\hat{v}_\phi(x) - v^*(x)\|_2 + \frac{R^2}{2}|\hat{L}-L|,$$

where we used $\|u\|_2 \le R$. This proves the lemma.

□

Notice that dictionary or proposal error does not appear in Lemma A.3; it is conceptually distinct from perturbing the quadratic objective itself. Instead, such error enters Theorem 4.1 through the quality of the best correction that remains available after proposal screening, relative to the full sparse correction class $\mathcal{U}_q$. Having separated these approximation effects, we next record the explicit form of the target descent direction used in the classification setting. For a linear softmax head, this identity follows directly from the cross-entropy gradient.

Lemma A.4. *Suppose $g_{\beta^*}(z) = Wz + b$ is a linear-logit multiclass head and $p(z) = \mathrm{softmax}(Wz + b)$. For a one-hot class vector e_y, cross-entropy satisfies $\nabla_z \ell(y, g_{\beta^*}(z)) = W^\top\{p(z) - e_y\}$. Consequently,*

$$v^*(x) = W^\top\{\eta(x) - p_0(x)\},$$

where $p_0(x) = p\{z_0(x)\}$. For observation i in cross-fitting fold k, the corresponding fold-specific pseudo-gradient is

$$G_i = \hat{W}^{(-k)\top}\left\{e_{Y_i} - \hat{p}_i^{(-k)}\right\}.$$

Proof. For logits $a(z) = Wz + b$ and class-probability vector $p(z) = \mathrm{softmax}\{a(z)\}$, multiclass cross-entropy for class y is $\ell(y, g_{\beta^*}(z)) = -\log p_y(z)$. Its gradient with respect to the logits is $\nabla_a \ell(y, g_{\beta^*}(z)) = p(z) - e_y$. Since $\nabla_z a(z) = W$, the chain rule gives

$$\nabla_z \ell(y, g_{\beta^*}(z)) = W^\top\{p(z) - e_y\}.$$

For P_0-almost every x, the conditional target risk can be written directly as $r_x(z) = -\sum_{c=1}^{C} \eta_c(x) \log p_c(z)$. Therefore, $\nabla r_x(z) = W^\top\{p(z) - \eta(x)\}$. Evaluating at $z = z_0(x)$ gives

$$\nabla r_x\{z_0(x)\} = W^\top\{p_0(x) - \eta(x)\},$$

and hence, by the definition $v^*(x) = -\nabla r_x\{z_0(x)\}$, $v^*(x) = W^\top\{\eta(x) - p_0(x)\}$.

Now let observation i belong to cross-fitting fold k. By definition, $G_i = -\nabla_z \ell\{Y_i, g_{\hat{\beta}^{(-k)}}(z_0(X_i))\}$. Applying the same cross-entropy gradient formula to the fold-specific fitted head yields

$$G_i = \hat{W}^{(-k)\top}\left\{e_{Y_i} - \hat{p}_i^{(-k)}\right\}.$$

□

This lemma explains why the learned object in (2.4) is a representation-space direction field, with no parameter-space target task vector involved. Cross-fitting ensures that the observation generating G_i was not used to fit the particular head appearing in its own pseudo-gradient; as stated in the main text, no orthogonal-score or double-machine-learning claim is required.

A.2 Directional Gain, Separation, and Safety Proofs

Proof of Theorem 4.1

Condition throughout on the fitted target-stage objects and on the frozen deployed specification $\hat{h}$. Thus $\hat{v}_\phi$, $\hat{L}$, the selected proposal mechanism, and the selected hyperparameters are fixed functions. All expectations below are with respect to a fresh target input $X \sim P_0$. Recall that the deployed correction is $\hat{u}(x) = \hat{\gamma} D_{\hat{k}}(x)\hat{\alpha}(x)$. Because $\hat{\alpha}(x) \geq 0$, $\|\hat{\alpha}(x)\|_0 \leq q$, $\|\hat{\alpha}(x)\|_1 \leq \hat{A}_{\max}$, and $0 \leq \hat{\gamma} \leq \gamma_{\max}$, its induced coefficients $\omega_e(x) = \hat{\gamma}\hat{\alpha}_e(x)$ satisfy

$$\omega(x) \geq 0, \quad \|\omega(x)\|_0 \leq q, \quad \|\omega(x)\|_1 \leq \gamma_{\max}\hat{A}_{\max} = C_{\max}.$$

Hence $\hat{u}(x) \in \mathcal{U}_q(x)$ for every x for which the deployed rule is defined, since its sparse coefficients satisfy all defining constraints. In particular, Lemma A.2 applies to $\hat{u}(x)$.

Let $\tilde{u}_q(x)$ be the measurable comparison correction appearing in Theorem 4.1, representing the best correction retained after proposal screening. By assumption,

$$\mathbb{E}_X\left[Q_X\{\tilde{u}_q(X)\} - Q_X\{u_q^*(X)\}\right] \leq \varepsilon_{\mathrm{dict}}. \tag{A.2}$$

For a finite source bank with deterministic tie-breaking, the required measurability is automatic from the same finite-support argument used in Lemma A.1. The term $\varepsilon_{\mathrm{dict}}$ therefore measures loss in population quadratic objective caused by directions that are removed or distorted before the composition step. The implemented correction need not minimize the unregularized estimated quadratic objective $\hat{Q}_x$, because the deployed procedure also includes ridge regularization, a finite amplitude grid, support screening, and numerical optimization. The theorem accordingly assumes

$$\hat{Q}_x\{\hat{u}(x)\} \le \hat{Q}_x\{\tilde{u}_q(x)\} + \epsilon_{\mathrm{alg}}(x), \quad \epsilon_{\mathrm{alg}}(x) \ge 0. \tag{A.3}$$

Applying Lemma A.3 to $\hat{u}(x)$ gives

$$\begin{aligned} Q_x\{\hat{u}(x)\} &\le \hat{Q}_x\{\hat{u}(x)\} + R\left\|\hat{v}_\phi(x) - v^*(x)\right\|_2 + \frac{R^2}{2}|\hat{L} - L| \\ &\le \hat{Q}_x\{\tilde{u}_q(x)\} + \epsilon_{\mathrm{alg}}(x) + R\left\|\hat{v}_\phi(x) - v^*(x)\right\|_2 + \frac{R^2}{2}|\hat{L} - L|, \end{aligned}$$

where the second inequality uses (A.3).

A second application of Lemma A.3, now to $\tilde{u}_q(x)$, yields

$$\begin{aligned} Q_x\{\hat{u}(x)\} &\le Q_x\{\tilde{u}_q(x)\} + 2R\left\|\hat{v}_\phi(x) - v^*(x)\right\|_2 + R^2|\hat{L} - L| + \epsilon_{\mathrm{alg}}(x) \\ &= Q_x\{u_q^*(x)\} + \left[Q_x\{\tilde{u}_q(x)\} - Q_x\{u_q^*(x)\}\right] \\ &\quad + 2R\left\|\hat{v}_\phi(x) - v^*(x)\right\|_2 + R^2|\hat{L} - L| + \epsilon_{\mathrm{alg}}(x). \end{aligned} \tag{A.4}$$

Since $\hat{u}(x) \in \mathcal{U}_q(x)$, Lemma A.2 gives

$$r_x\{z_0(x) + \hat{u}(x)\} - r_x\{z_0(x)\} \le Q_x\{\hat{u}(x)\}.$$

Combining this inequality with (A.4) and taking expectations over a fresh $X \sim P_0$ gives

$$\begin{aligned} &\mathbb{E}_X\left[r_X\{z_0(X) + \hat{u}(X)\} - r_X\{z_0(X)\}\right] \\ &\le \mathbb{E}_X Q_X\{u_q^*(X)\} + \varepsilon_{\mathrm{dict}} + 2R\mathbb{E}_X\left\|\hat{v}_\phi(X) - v^*(X)\right\|_2 + R^2|\hat{L} - L| + \mathbb{E}_X\epsilon_{\mathrm{alg}}(X), \end{aligned}$$

where (A.2) was used for the dictionary term.

Define, as in the theorem, $\varepsilon_v = \mathbb{E}_X\left\|\hat{v}_\phi(X) - v^*(X)\right\|_2$, $\varepsilon_L = R^2|\hat{L} - L|$, and $\varepsilon_{\mathrm{alg}} = \mathbb{E}_X\epsilon_{\mathrm{alg}}(X)$. Because we have conditioned on the fitted target-stage objects, $\hat{L}$ is fixed, so no additional expectation appears in ε_L. Finally, $\Gamma_q = -\mathbb{E}_X Q_X\{u_q^*(X)\}$, and therefore

$$\mathbb{E}_X\left[r_X\{z_0(X) + \hat{u}(X)\} - r_X\{z_0(X)\}\right] \le -\Gamma_q + 2R\varepsilon_v + \varepsilon_{\mathrm{dict}} + \varepsilon_L + \varepsilon_{\mathrm{alg}},$$

which proves (4.4).

□

The preceding theorem is conditional and remains valid for any realized field error ε_v. We next give one explicit set of conditions under which that error decreases with the amount of target training information. Let $\mathcal{I}_1, \ldots, \mathcal{I}_{K_{\mathrm{cf}}}$ denote the cross-fitting folds, let $n_k = |\mathcal{I}_k|$, and put $\pi_k = n_k/n_{\mathrm{tr}}$. For $i \in \mathcal{I}_k$, condition on the observations outside fold k used to construct the fold-specific target head and define $m_k(x) = \mathbb{E}(G_i \mid X_i = x, \mathcal{D}_0^{\mathrm{tr}} \setminus \mathcal{I}_k)$. The pooled cross-fitted regression target is

$$m_{\mathrm{cf}}(x) = \sum_{k=1}^{K_{\mathrm{cf}}} \pi_k m_k(x).$$

Thus m_{cf} is the conditional mean function implicitly targeted by least-squares regression on the pooled pseudo-gradients. Relating the fitted field to the population target direction v^* requires two distinct controls; the first measures the discrepancy between this cross-fitted regression target and v^*, while the second controls statistical estimation within the chosen direction-field class. Assumptions 4 and 5 formalize these two ingredients, respectively.

Assumption 4. *For some $\varepsilon_{\mathrm{cf}} \ge 0$, $\|m_{\mathrm{cf}} - v^*\|_{L_2(P_0)} \le \varepsilon_{\mathrm{cf}}$. The candidate field classes form a finite training-only library $\mathfrak{H}_v$. For every class $\mathcal{V} \in \mathfrak{H}_v$, the functions are uniformly bounded in Euclidean norm, and for every empirical probability measure Q,*

$$\log N\{\epsilon, \mathcal{V}, L_2(Q)\} \le V_v \log(A_v/\epsilon)$$

over the relevant range of ϵ, with common finite constants V_v, A_v. The pseudo-gradients are uniformly bounded in norm.

The quantity $\varepsilon_{\mathrm{cf}}$ isolates error because the cross-fitted target heads approximate but do not equal the population head. It does not presume an orthogonal moment condition. If the fold-specific target heads are consistent strongly enough that $m_{\mathrm{cf}} \to v^*$ in $L_2(P_0)$, then $\varepsilon_{\mathrm{cf}} \to 0$. To state a transparent fast-rate condition, define

$$\hat{\mathcal{R}}(v) = \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \|G_i - v(X_i)\|_2^2$$

and let $\mathcal{R}_{\mathrm{cf}}(v)$ be the corresponding fold-weighted conditional population risk. Let $v^\dagger = v_{\phi^\dagger}$ be a comparator in the selected class and define $a_v = \left\|v^\dagger - v^*\right\|_{L_2(P_0)}$ and $J_v = \left\|\phi^\dagger\right\|_2^2$.

Assumption 5. *For a failure probability $\delta_v \in (0,1)$, define*

$$\rho_n = \sqrt{\frac{V_v \log(A_v n_{\mathrm{tr}}) + \log\{2|\mathfrak{H}_v|K_{\mathrm{cf}}/\delta_v\}}{n_{\mathrm{tr}}}}.$$

With probability at least $1 - \delta_v$, simultaneously for every field class in the finite training-only library and every v in that class,

$$\left|\{\hat{\mathcal{R}}(v) - \hat{\mathcal{R}}(v^\dagger)\} - \{\mathcal{R}_{\mathrm{cf}}(v) - \mathcal{R}_{\mathrm{cf}}(v^\dagger)\}\right| \le c_0 \rho_n \left\|v - v^\dagger\right\|_{L_2(P_0)} + c_0 \rho_n^2 \tag{A.5}$$

for a constant c_0 independent of n_{tr}.

Condition (A.5) is the usual localized, or fast-rate, square-loss empirical-process condition. For bounded finite-dimensional or VC-type field classes it follows from foldwise concentration and localization of squared-loss excess risk. The cross-fitting structure is compatible with the condition because each held-out fold is independent of the head used to construct its own pseudo-gradients; a union bound handles the finite number of folds and the finite training-only field library. Neural direction fields may require a class-specific complexity analysis, which is why the main theorem is intentionally stated directly in terms of ε_v. Proposition A.5 then translates the empirical least-squares fit of the direction field into an explicit population $L_2(P_0)$ error bound, and hence into a rate for ε_v.

Proposition A.5. *Suppose Assumptions 4-5 hold. Assume $\hat{\phi}$ satisfies the penalized empirical comparison*

$$\hat{\mathcal{R}}(v_{\hat{\phi}}) + \lambda_v \left\|\hat{\phi}\right\|_2^2 \le \hat{\mathcal{R}}(v_{\phi^\dagger}) + \lambda_v \left\|\phi^\dagger\right\|_2^2 + \xi_v$$

for some optimization tolerance $\xi_v \ge 0$. Then, on the event in Assumption 5,

$$\left\|\hat{v}_\phi - v^*\right\|_{L_2(P_0)} \le C\left\{a_v + \varepsilon_{\mathrm{cf}} + \rho_n + \sqrt{\lambda_v J_v + \xi_v}\right\},$$

where C depends only on the constant in (A.5). Consequently,

$$\varepsilon_v = \mathbb{E}_X \left\|\hat{v}_\phi(X) - v^*(X)\right\|_2 \le \left\|\hat{v}_\phi - v^*\right\|_{L_2(P_0)}$$

obeys the same bound.

Proof. Condition on the fold-specific fitted heads used to construct the cross-fitted pseudo-gradients. For fold k, let $m_k(x) = \mathbb{E}[G_i \mid X_i = x,\ i \in I_k]$, and write

$$m_{\mathrm{cf}}(x) = \sum_{k=1}^{K_{\mathrm{cf}}} \pi_k m_k(x), \quad \pi_k = \frac{|I_k|}{n_{\mathrm{tr}}}.$$

For any measurable field v, the conditional squared-loss identity gives

$$\mathbb{E}\left[\|v(X) - G_i\|_2^2 \mid i \in I_k\right] = \mathbb{E}_X \|v(X) - m_k(X)\|_2^2 + C_k,$$

where C_k does not depend on v. Hence

$$\sum_{k=1}^{K_{\mathrm{cf}}} \pi_k \mathbb{E}_X \left\| v(X) - m_k(X) \right\|_2^2 = \mathbb{E}_X \left\| v(X) - m_{\mathrm{cf}}(X) \right\|_2^2 + \sum_{k=1}^{K_{\mathrm{cf}}} \pi_k \mathbb{E}_X \left\| m_k(X) - m_{\mathrm{cf}}(X) \right\|_2^2 . \tag{A.6}$$

The second term in (A.6) is independent of v. Consequently, excess population squared loss for the pooled cross-fitted regression problem is exactly $\|v - m_{\mathrm{cf}}\|_{L_2(P_0)}^2$.

Let $v^\dagger$ be the comparison field supplied by Assumption 4, so that $\left\|v^\dagger - m_{\mathrm{cf}}\right\|_{L_2(P_0)} \le a_v$. The approximate penalized empirical optimality of $\hat{v}_\phi$ gives

$$\hat{\mathcal{R}}(\hat{v}_\phi) + \lambda_v \mathcal{J}_v(\hat{v}_\phi) \le \hat{\mathcal{R}}(v^\dagger) + \lambda_v \mathcal{J}_v(v^\dagger) + \xi_v,$$

where $J_v(v^\dagger) = j_v$. Since $\mathcal{J}_v(\hat{v}_\phi) \ge 0$, subtracting the corresponding population risks and applying Assumption 5 yields, on its stated high-probability event,

$$\|\hat{v}_\phi - m_{\mathrm{cf}}\|_{L_2(P_0)}^2 \le a_v^2 + C_1 \rho_n \left\|\hat{v}_\phi - v^\dagger\right\|_{L_2(P_0)} + C_2 \rho_n^2 + \lambda_v \mathcal{J}_v(v^\dagger) + \xi_v,$$

for universal constants $C_1, C_2 > 0$.

By the triangle inequality,

$$\left\|\hat{v}_\phi - v^\dagger\right\|_{L_2(P_0)} \le \|\hat{v}_\phi - m_{\mathrm{cf}}\|_{L_2(P_0)} + a_v.$$

Writing $t = \|\hat{v}_\phi - m_{\mathrm{cf}}\|_{L_2(P_0)}$ and substituting the preceding bound gives

$$t^2 \le a_v^2 + C_1 \rho_n (t + a_v) + C_2 \rho_n^2 + \lambda_v \mathcal{J}_v(v^\dagger) + \xi_v.$$

Using $2ab \le a^2 + b^2$ to absorb the term proportional to $\rho_n t$ into the left-hand side, and enlarging the universal constant if necessary, gives

$$\|\hat{v}_\phi - m_{\mathrm{cf}}\|_{L_2(P_0)} \le C \left\{ a_v + \rho_n + \sqrt{\lambda_v \mathcal{J}_v(v^\dagger) + \xi_v} \right\}.$$

Assumption 4 also gives $\|m_{\mathrm{cf}} - v^*\|_{L_2(P_0)} \le \varepsilon_{\mathrm{cf}}$. A final application of the triangle inequality therefore yields

$$\|\hat{v}_\phi - v^*\|_{L_2(P_0)} \le C \left\{ a_v + \varepsilon_{\mathrm{cf}} + \rho_n + \sqrt{\lambda_v \mathcal{J}_v(v^\dagger) + \xi_v} \right\},$$

after enlarging C if needed.

Finally, by Cauchy-Schwarz,

$$\mathbb{E}_X \left\| \hat{v}_\phi(X) - v^*(X) \right\|_2 \le \left\{ \mathbb{E}_X \left\| \hat{v}_\phi(X) - v^*(X) \right\|_2^2 \right\}^{1/2},$$

so the same bound controls the field-error quantity ε_v appearing in Theorem 4.1.

□

For a well-specified finite-complexity field with $a_v = \varepsilon_{\mathrm{cf}} = \xi_v = 0$ and sufficiently small $\lambda_v J_v$, Proposition A.5 gives the familiar

$$\varepsilon_v = O_{\mathbb{P}} \left[\sqrt{\frac{V_v \log(A_v n_{\mathrm{tr}}) + \log(1/\delta_v)}{n_{\mathrm{tr}}}} \right]$$

up to the fixed fold/library multiplicity. Substituting this rate into Theorem 4.1 shows explicitly why increasing target information can make the local directional gain dominate direction-field estimation error. This is the theoretical mechanism examined in the target-sample-size experiment of Section 5.

PROOF OF THEOREM 4.2

Let $X \in \{G_1, G_2\}$ with $\mathbb{P}(X \in G_1) = \mathbb{P}(X \in G_2) = \frac{1}{2}$, and consider scalar prediction with conditional squared risk $r_x(u) = \frac{1}{2}(1-u)^2$. The base correction is zero, and the two source functional directions satisfy

$$d_1(G_1) = 1,\ d_1(G_2) = -1,\ \text{and}\ d_2(G_1) = -1,\ d_2(G_2) = 1.$$

Consider first any rule with input-independent coefficients $(a_1, a_2) \in \mathbb{R}^2$. Writing $c = a_1 - a_2$, its correction equals c on G_1 and $-c$ on G_2. Its population risk is therefore

$$\frac{1}{2}\left\{\frac{1}{2}(1-c)^2\right\} + \frac{1}{2}\left\{\frac{1}{2}(1+c)^2\right\} = \frac{1}{4}\{(1-c)^2 + (1+c)^2\} = \frac{1}{2}(1+c^2) \geq \frac{1}{2}.$$

This lower bound holds for every $(a_1, a_2) \in \mathbb{R}^2$. By contrast, the input-dependent rule that selects source 1 on G_1 and source 2 on G_2 produces correction one on both regions. Its population risk is therefore zero. Hence every input-independent rule has excess risk at least $1/2$ relative to this local oracle. The two source directions also have identical global average directional alignment with the desired constant correction. Indeed,

$$\mathbb{E}_X d_1(X) = \frac{1}{2}(1) + \frac{1}{2}(-1) = 0 \text{ and } \mathbb{E}_X d_2(X) = \frac{1}{2}(-1) + \frac{1}{2}(1) = 0.$$

Thus a global average directional criterion cannot distinguish the two sources in this construction.

It remains to establish the stated perturbation claim. We fix $A_g < \infty$ and restrict the input-independent coefficients to the compact set

$$\mathcal{A}_g = \left\{(a_1, a_2) \in \mathbb{R}^2 : |a_1| + |a_2| \leq A_g\right\}.$$

Consider perturbed region probabilities $\tilde{p}_r$, regression targets $\tilde{m}_r$, and functional directions $\tilde{d}_e(G_r)$ satisfying

$$\max_{r=1,2}\left|\tilde{p}_r - \frac{1}{2}\right| \leq \epsilon,\ \max_{r=1,2}|\tilde{m}_r - 1| \leq \epsilon,\ \text{and}\ \max_{e,r}|\tilde{d}_e(G_r) - d_e(G_r)| \leq \epsilon,$$

with $\tilde{p}_1 + \tilde{p}_2 = 1$. For $a = (a_1, a_2) \in \mathcal{A}_g$, let

$$R_\epsilon^{\mathrm{glob}}(a) = \frac{1}{2}\sum_{r=1}^{2} \tilde{p}_r \left\{\tilde{m}_r - a_1\tilde{d}_1(G_r) - a_2\tilde{d}_2(G_r)\right\}^2.$$

Because $\mathcal{A}_g$ is compact and all perturbed quantities remain uniformly bounded for sufficiently small ϵ, the squared-loss criterion is uniformly Lipschitz in these quantities on the resulting bounded set. Consequently, for a constant $C_g < \infty$ depending only on A_g,

$$\sup_{a \in \mathcal{A}_g}\left|R_\epsilon^{\mathrm{glob}}(a) - R_0^{\mathrm{glob}}(a)\right| \leq C_g\epsilon.$$

Since $\inf_{a\in\mathcal{A}_g} R_0^{\mathrm{glob}}(a) \geq \frac{1}{2}$, it follows that

$$\inf_{a \in \mathcal{A}_g} R_\epsilon^{\mathrm{glob}}(a) \geq \frac{1}{2} - C_g\epsilon.$$

Now consider the perturbed local rule that uses source 1 with unit coefficient on G_1 and source 2 with unit coefficient on G_2. Its risk satisfies

$$R_\epsilon^{\mathrm{loc}} = \frac{1}{2}\left[\tilde{p}_1\{\tilde{m}_1 - \tilde{d}_1(G_1)\}^2 + \tilde{p}_2\{\tilde{m}_2 - \tilde{d}_2(G_2)\}^2\right].$$

Since each difference inside braces has absolute value at most 2ϵ, $R_\epsilon^{\mathrm{loc}} \leq 2\epsilon^2$. Therefore

$$\inf_{a \in \mathcal{A}_g} R_\epsilon^{\mathrm{glob}}(a) - R_\epsilon^{\mathrm{loc}} \geq \frac{1}{2} - C_g\epsilon - 2\epsilon^2.$$

For all sufficiently small $\epsilon > 0$, the right-hand side is bounded away from zero; for example, it exceeds $1/4$ after decreasing the perturbation radius if necessary. Thus a positive constant local-global separation persists uniformly over any fixed compact set of input-independent coefficients.

□

The exact construction above allows unrestricted global coefficients. The following lemma makes precise the robustness statement under perturbation.

Lemma A.6. *We fix any $A_g < \infty$ and restrict input-independent coefficients to the compact set $\mathcal{A}_g = \{(a_1, a_2) \in \mathbb{R}^2 : |a_1| + |a_2| \le A_g\}$. Let $\mathcal{I}_g$ be a fixed compact interval containing all global corrections generated by $\mathcal{A}_g$ and all local corrections considered below. Suppose the perturbed region probabilities, source directions, and conditional risks satisfy*

$$\left|\mathbb{P}(X \in G_j) - \frac{1}{2}\right| \le \epsilon, \qquad \max_{j\in\{1,2\},\, e\in\{1,2\}} \left|\tilde{d}_e(G_j) - d_e(G_j)\right| \le \epsilon,$$

and

$$\sup_{j\in\{1,2\},\, u\in\mathcal{I}_g} \left|\tilde{r}_{G_j}(u) - \frac{1}{2}(1-u)^2\right| \le \epsilon.$$

Then there exist constants $\epsilon_0 > 0$ and $c_0 > 0$, depending only on A_g and $\mathcal{I}_g$, such that, for $0 < \epsilon \le \epsilon_0$, the best perturbed input-independent rule has risk at least c_0 more than the perturbed local rule that uses source 1 on G_1 and source 2 on G_2.

Proof. We write $q(u) = \frac{1}{2}(1-u)^2$ and $p_j = \mathbb{P}(X \in G_j)$, and let $L_g = \sup_{u\in\mathcal{I}_g} |q'(u)| = \sup_{u\in\mathcal{I}_g} |1-u| < \infty$. We fix $a = (a_1, a_2) \in \mathcal{A}_g$ and define its perturbed global correction on region G_j by $\tilde{u}_j(a) = a_1\tilde{d}_1(G_j) + a_2\tilde{d}_2(G_j)$. In the unperturbed construction, writing $c = a_1 - a_2$, the corresponding corrections are $u_1(a) = c$ and $u_2(a) = -c$. Since $|a_1| + |a_2| \le A_g$,

$$|\tilde{u}_j(a) - u_j(a)| \le (|a_1| + |a_2|)\epsilon \le A_g\epsilon.$$

Because q is L_g-Lipschitz on $\mathcal{I}_g$,

$$q\{\tilde{u}_j(a)\} \ge q\{u_j(a)\} - L_gA_g\epsilon.$$

The assumed perturbation of the conditional risks therefore gives

$$\tilde{r}_{G_j}\{\tilde{u}_j(a)\} \ge q\{u_j(a)\} - (1 + L_gA_g)\epsilon.$$

Consequently, the perturbed population risk of the global rule satisfies

$$\tilde{R}^{\mathrm{glob}}(a) = \sum_{j=1}^{2} p_j\tilde{r}_{G_j}\{\tilde{u}_j(a)\} \ge p_1q(c) + p_2q(-c) - (1 + L_gA_g)\epsilon.$$

Since $p_1 + p_2 = 1$ and $|p_1 - \frac{1}{2}| \le \epsilon$,

$$p_1q(c) + p_2q(-c) = \frac{1}{2}\{q(c) + q(-c)\} + \left(p_1 - \frac{1}{2}\right)\{q(c) - q(-c)\}.$$

Now

$$\frac{1}{2}\{q(c) + q(-c)\} = \frac{1}{2}(1 + c^2) \ge \frac{1}{2},$$

while $q(c) - q(-c) = -2c$. Because $|c| \le A_g$,

$$p_1q(c) + p_2q(-c) \ge \frac{1}{2} - 2A_g\epsilon.$$

Hence, uniformly over $a \in \mathcal{A}_g$,

$$\tilde{R}^{\mathrm{glob}}(a) \ge \frac{1}{2} - C_g\epsilon, \quad C_g = 1 + A_g(L_g + 2).$$

It follows that

$$\inf_{a\in\mathcal{A}_g} \tilde{R}^{\mathrm{glob}}(a) \ge \frac{1}{2} - C_g\epsilon.$$

Now consider the perturbed local rule that uses source 1 with unit coefficient on G_1 and source 2 with unit coefficient on G_2. Its regional corrections are $\tilde{u}_1^{\mathrm{loc}} = \tilde{d}_1(G_1)$ and $\tilde{u}_2^{\mathrm{loc}} = \tilde{d}_2(G_2)$. Since $d_1(G_1) = d_2(G_2) = 1$, we have $|\tilde{u}_j^{\mathrm{loc}} - 1| \le \epsilon,\ j = 1, 2$. Therefore

$$q(\tilde{u}_j^{\mathrm{loc}}) = \frac{1}{2}(1 - \tilde{u}_j^{\mathrm{loc}})^2 \le \frac{1}{2}\epsilon^2,$$

and the conditional-risk perturbation yields

$$\tilde{r}_{G_j}(\tilde{u}_j^{\mathrm{loc}}) \le \epsilon + \frac{1}{2}\epsilon^2.$$

Thus the perturbed local risk obeys

$$\tilde{R}^{\mathrm{loc}} \le \epsilon + \frac{1}{2}\epsilon^2.$$

Combining the preceding bounds gives

$$\inf_{a \in \mathcal{A}_g} \tilde{R}^{\mathrm{glob}}(a) - \tilde{R}^{\mathrm{loc}} \ge \frac{1}{2} - (C_g + 1)\epsilon - \frac{1}{2}\epsilon^2.$$

The right-hand side converges to $1/2$ as $\epsilon \downarrow 0$. Hence there exists $\epsilon_0 > 0$ such that, for every $0 < \epsilon \le \epsilon_0$,

$$\inf_{a \in \mathcal{A}_g} \tilde{R}^{\mathrm{glob}}(a) - \tilde{R}^{\mathrm{loc}} \ge \frac{1}{4}.$$

Taking $c_0 = 1/4$ proves the result.

□

The compactness qualification in Lemma A.6 is important. Without it, an arbitrarily small generic perturbation can make the two regionwise source-direction vectors linearly independent, after which unbounded global coefficients could exploit the perturbation. The exact unperturbed separation in Theorem 4.2, by contrast, holds for all $a_1, a_2 \in \mathbb{R}$.

PROOF OF PROPOSITION 4.3

Condition on $\mathcal{D}_0^{\mathrm{tr}}$, all algorithmic randomness used during training, and the resulting finite candidate family $\{f_h : h \in \Lambda\}$. The candidates are therefore fixed before $\mathcal{D}_0^{\mathrm{val}}$ is opened. For each fixed $h \in \Lambda$, define $Z_{ih} = \ell_B\{Y_i, f_h(X_i)\} - \ell_B\{Y_i, f_0(X_i)\}$. Because $0 \le \ell_B \le B$, $-B \le Z_{ih} \le B$. Conditional on the fitted candidate family, the validation observations are independent and identically distributed (i.i.d.) from P_0, and hence $Z_{1h}, \ldots, Z_{n_{\mathrm{val}},h}$ are i.i.d. with

$$\mathbb{E}Z_{ih} = R_B(f_h) - R_B(f_0).$$

For a fixed h, set

$$Y_{ih} = \frac{Z_{ih} + B}{2B} \in [0, 1].$$

Its sample mean and unbiased sample variance satisfy

$$\bar{Y}_h = \frac{\bar{Z}_h + B}{2B} \text{ and } \hat{s}_{Y,h}^2 = \frac{\hat{s}_h^2}{4B^2}.$$

Applying the empirical-Bernstein inequality of Maurer and Pontil (2009) with failure probability $\eta \in (0, 1)$ gives

$$\mathbb{P}\left\{ \mathbb{E}Y_{ih} > \bar{Y}_h + \sqrt{\frac{2\hat{s}_{Y,h}^2 \log(2/\eta)}{n_{\mathrm{val}}}} + \frac{7\log(2/\eta)}{3(n_{\mathrm{val}} - 1)} \right\} \le \eta.$$

Multiplying by $2B$ and subtracting B yields

$$\mathbb{P}\left\{ R_B(f_h) - R_B(f_0) > \bar{Z}_h + \sqrt{\frac{2\hat{s}_h^2 \log(2/\eta)}{n_{\mathrm{val}}}} + \frac{14B\log(2/\eta)}{3(n_{\mathrm{val}} - 1)} \right\} \le \eta.$$

Set $\eta = \frac{\delta}{|\Lambda|}$ and $L_\delta = \log \frac{2|\Lambda|}{\delta}$. Then, for every fixed $h \in \Lambda$,

$$\mathbb{P}\{R_B(f_h) - R_B(f_0) > U_h\} \le \frac{\delta}{|\Lambda|}.$$

A union bound over the finite candidate family therefore gives an event $\mathcal{E}_{\mathrm{val}}$ with $\mathbb{P}(\mathcal{E}_{\mathrm{val}}) \geq 1-\delta$ on which

$$R_B(f_h) - R_B(f_0) \leq U_h \quad \text{simultaneously for every } h \in \Lambda.$$

If LTD-SAFE selects a transfer candidate $\hat{h}$, then by its certification rule $U_{\hat{h}} \leq 0$. Therefore, on $\mathcal{E}_{\mathrm{val}}$,

$$R_B(f_{\hat{h}}) - R_B(f_0) \leq U_{\hat{h}} \leq 0,$$

and hence

$$R_B(f_{\hat{h}}) \leq R_B(f_0).$$

If no transfer candidate is certified, LTD-SAFE returns f_0 exactly, so equality holds.

Finally, because $\mathcal{E}_{\mathrm{val}}$ controls every $h \in \Lambda$ simultaneously, $\hat{h}$ may be chosen using the same validation sample from among the certified candidates without any additional selection correction. This proves the proposition.

□

A.3 CONNECTION TO PARAMETER ANGLES

PROOF OF PROPOSITION 2.1

A source update Δ_s changes the linear prediction at input x by $d_s(x) = f_{\beta+\Delta_s}(x) - f_\beta(x) = x^\top \Delta_s$. Likewise, the target correction Δ_0 induces $d_0(x) = x^\top \Delta_0$. Therefore

$$\mathbb{E}\{d_s(X)d_0(X)\} = \mathbb{E}\left[(X^\top \Delta_s)(X^\top \Delta_0)\right] = \mathbb{E}\left[\Delta_s^\top X X^\top \Delta_0\right] = \Delta_s^\top \mathbb{E}(XX^\top)\Delta_0 = \Delta_s^\top \Sigma_X \Delta_0.$$

If $\Sigma_X = I$, then

$$\mathbb{E}\{d_s(X)^2\} = \Delta_s^\top \Delta_s = \|\Delta_s\|_2^2,$$

and similarly

$$\mathbb{E}\{d_0(X)^2\} = \|\Delta_0\|_2^2.$$

Hence, for nonzero Δ_s and Δ_0, normalized functional alignment is

$$\frac{\mathbb{E}\{d_s(X)d_0(X)\}}{\{\mathbb{E}d_s(X)^2\}^{1/2}\{\mathbb{E}d_0(X)^2\}^{1/2}} = \frac{\Delta_s^\top \Delta_0}{\|\Delta_s\|_2 \|\Delta_0\|_2},$$

which is the ordinary parameter cosine.

More generally, if Σ_X is positive definite, define $\langle a, b\rangle_{\Sigma_X} = a^\top \Sigma_X b$ and $\|a\|_{\Sigma_X} = (a^\top \Sigma_X a)^{1/2}$. Then, for nonzero Δ_s and Δ_0,

$$\frac{\mathbb{E}\{d_s(X)d_0(X)\}}{\{\mathbb{E}d_s(X)^2\}^{1/2}\{\mathbb{E}d_0(X)^2\}^{1/2}} = \frac{\langle \Delta_s, \Delta_0\rangle_{\Sigma_X}}{\|\Delta_s\|_{\Sigma_X} \|\Delta_0\|_{\Sigma_X}},$$

which is precisely the cosine in the inner-product geometry induced by Σ_X.

□

Proposition 2.1 identifies the linear setting in which a parameter angle has a direct functional interpretation. In that case, population alignment is obtained by averaging source and target prediction changes over the target input distribution, with geometry determined by $\Sigma_X = \mathbb{E}(XX^\top)$. In a nonlinear network, however, the finite functional direction $d_e(x)$ can vary substantially with x, so a single population average can obscure directions that are beneficial on some target inputs and harmful on others. Theorem 4.2 isolates this loss of local information in a simple exact construction.

The theoretical results are stated directly on the representation scale. They use the exact finite changes

$$d_e(x) = z_{\theta_0 + P_b \Delta_s}(x) - z_{\theta_0}(x),$$

without any parameter-space linearization, and the gain bound applies only over the bounded representation neighborhood specified in Assumption 2. Theorem 4.1 characterizes when a learned local correction approaches a useful population directional oracle, whereas Proposition 4.3 gives the separate finite-sample target-only protection for the prespecified bounded validation loss.

B Optimization and Algorithmic Details

This section specifies the construction of the source-block bank, estimation of the target direction field and proposal scorer, and the sparse composition rule used after proposal. All choices described here use only the source checkpoints and $\mathcal{D}_0^{\mathrm{tr}}$; target validation is opened only after the resulting candidate predictors have been frozen.

B.1 Source Blocks and Exact Functional Directions

For source s, let $\Delta_s = \theta_s - \theta_0$. We first fix a disjoint architectural partition $\mathcal{B}_s^{\mathrm{all}}$ of the update coordinates. Writing P_b for the coordinate projection associated with block b, the partition satisfies

$$P_b P_{b'} = 0 \text{ for } b \neq b', \quad \Delta_s = \sum_{b \in \mathcal{B}_s^{\mathrm{all}}} P_b \Delta_s.$$

The source-block bank used by LTD is a prespecified subset $\mathcal{B}_s \subseteq \mathcal{B}_s^{\mathrm{all}}$, and $\mathcal{E} = \{(s, b) : s \in [S],\ b \in \mathcal{B}_s\}$. Thus restricting the compute-aware bank does not imply that the retained blocks themselves form a complete decomposition of Δ_s.

For a ViT with L transformer blocks, the compute-aware default retains the latter half of the backbone, $\{\lfloor L/2 \rfloor + 1, \ldots, L\}$, divides these layers into three contiguous groups whose sizes differ by at most one, and treats the final normalization/projection as a fourth block. The full-depth ablation instead partitions layers $1, \ldots, L$ into four contiguous groups of sizes differing by at most one while retaining the same final normalization/projection component. For LoRA or adapter models, all trainable adapter tensors attached to one transformer layer form one block. For residual CNNs, the final two residual stages and the projection head form the default bank. These rules are fixed before target adaptation and are unchanged across target splits and random seeds.

For $e = (s, b) \in \mathcal{E}$, define $\theta_e = \theta_0 + P_b \Delta_s$. At the designated fusion representation, $d_e(x) = z_{\theta_e}(x) - z_0(x)$ is evaluated by an actual hybrid forward computation; no Jacobian-vector approximation is used. The location and preprocessing of the fusion representation are fixed before validation and are applied identically to the base and hybrid checkpoints. In particular, if the architecture exposes a normalized output at the designated fusion layer, that normalized output is taken as $z_\theta(x)$. For CLIP, we use the usual unit-normalized embedding, so

$$d_e(x) = \frac{h_{\theta_e}(x)}{\|h_{\theta_e}(x)\|_2} - \frac{h_{\theta_0}(x)}{\|h_{\theta_0}(x)\|_2},$$

where $h_\theta(x)$ denotes the corresponding pre-normalization embedding. This remains an exact finite difference in the designated representation space. A tangent-space projection, when reported, is treated as a separate ablation and is not substituted for d_e in the primary method or theory.

For architectures exposing block-boundary activations, one base forward pass caches the input activation to every eligible source block. Evaluation of expert $e = (s, b)$ then reuses the cached prefix, executes the modified block, and recomputes the downstream base suffix. We define F_e to be the incremental FLOPs of this complete branch computation, the modified block together with its required downstream suffix, conditional on the cached base prefix. Thus F_e does not exclude computation required to execute the source block itself. If a software stack does not expose safe block-boundary reuse, the corresponding hybrid direction is evaluated by a full hybrid forward pass and is charged its actual measured cost, with the idealized suffix cost excluded. Persistent source-update storage and peak execution memory are reported separately from FLOPs.

B.2 Direction-Field and Proposal Learning

We partition $\mathcal{D}_0^{\mathrm{tr}}$ into $K_{\mathrm{cf}} = 5$ fixed folds $I_1, \ldots, I_{K_{\mathrm{cf}}}$. For each fold k, the target head is fitted on $\mathcal{D}_0^{\mathrm{tr}} \setminus I_k$, and for $i \in I_k$ we form

$$G_i = -\nabla_z \ell\Big\{Y_i, g_{\hat{\beta}^{(-k)}}(z_0(X_i))\Big\}.$$

Hence observation i is not used to fit the head that generates its own pseudo-gradient.

The candidate direction-field classes are a constant field, a linear field, and two small MLP fields. For the latter we use

$$v_\phi(x) = W_2\sigma\{W_1 z_0(x) + b_1\} + b_2,$$

with output dimension d, fixed activation σ, and hidden width $d/16$ or $d/8$ up to integer rounding. For any candidate class and λ_v, the fitted field minimizes

$$\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}} \|v_\phi(X_i) - G_i\|_2^2 + \lambda_v \|\phi\|_2^2 .$$

The field class and λ_v are selected using a fixed inner split of the cross-fitted pairs $\{(X_i, G_i) : i \in [n_{\mathrm{tr}}]\}$, with mean squared pseudo-gradient prediction error as the selection criterion. The inner split is contained entirely in $\mathcal{D}_0^{\mathrm{tr}}$. After selection, the chosen field class is refitted on all n_{tr} cross-fitted pairs to produce $\hat{v}_\phi$. Ties are resolved by a fixed deterministic rule favoring the lower-complexity field and then the larger value of λ_v. Neither field selection nor refitting uses $\mathcal{D}_0^{\mathrm{val}}$. After all pseudo-gradients have been constructed, the target prediction head is refitted separately on the full target training sample. Thus the head deployed for prediction may use all of $\mathcal{D}_0^{\mathrm{tr}}$, whereas the pseudo-gradients used to train the direction field remain cross-fitted.

For proposal learning and each candidate curvature value $\hat{L}$, define the cached corrected-gain target

$$\tilde{S}_{ie}^{(\hat{L})} = \langle G_i, d_e(X_i)\rangle - \frac{\hat{L}}{2}\|d_e(X_i)\|_2^2 .$$

For each i, order the M experts by $\tilde{S}_{ie}^{(\hat{L})}$, breaking ties by their fixed expert index, and let

$$m_+ = \max\left\{1, \left\lceil \frac{M}{4} \right\rceil\right\}.$$

The upper m_+ experts form the positive pool $\mathcal{E}_{i,+}^{(\hat{L})}$. One positive e_i^+ is sampled uniformly from this pool using the fixed proposal-training seed. Among the remaining experts, the four with largest corrected gain (or all remaining experts when fewer than four are available) form the near-boundary hard-negative set $\mathcal{E}_{i,-}^{(\hat{L})}$. The resulting pair set is

$$\mathcal{P}_{\hat{L}} = \left\{(i, e_i^+, e^-) : i \in [n_{\mathrm{tr}}],\ e^- \in \mathcal{E}_{i,-}^{(\hat{L})}\right\}.$$

Once formed, this pair set remains fixed throughout optimization. The proposal scorer has the form in (3.1). For curvature value $\hat{L}$, its parameters minimize

$$\hat{\mathcal{L}}_{\mathrm{rank}}^{(\hat{L})} = \frac{1}{|\mathcal{P}_{\hat{L}}|}\sum_{(i,e^+,e^-)\in\mathcal{P}_{\hat{L}}} \log\left[1 + \exp\left\{-r_{e^+}^{(\hat{L})}(X_i) + r_{e^-}^{(\hat{L})}(X_i)\right\}\right].$$

Thus the proposal learner is trained only to reproduce the ordering induced by the cross-fitted corrected-gain targets; it does not itself determine whether transfer is beneficial.

The expert descriptor contains only checkpoint-side information. We use a learned expert-identity embedding and a learned source-identity embedding together with normalized block depth and the block RMS update magnitude

$$\nu_e = \frac{\|P_b\Delta_s\|_2}{\sqrt{\dim(P_b\Delta_s)}}.$$

Their concatenation is projected to the fixed descriptor dimension d_u and constitutes u_e. No source examples are required. Distinct values of $\hat{L}$ use distinct proposal parameters, including their expert embeddings, while sharing the cached quantities $z_0(X_i),\ G_i,\ d_e(X_i),\ \langle G_i, d_e(X_i)\rangle$, and $\|d_e(X_i)\|_2^2$. All proposal heads are fitted using $\mathcal{D}_0^{\mathrm{tr}}$ only and are frozen before target validation.

B.3 Sparse Composition and Deployment Cost

For a test input x, let $J_k(x) = (e_1, \ldots, e_k)$ be the proposal shortlist and $D_k(x) = [d_{e_1}(x), \ldots, d_{e_k}(x)] \in \mathbb{R}^{d\times k}$. The deployed sparse composition obeys the same coefficient constraints as the optimization problem in the main text, $\alpha \geq 0,\ \|\alpha\|_0 \leq q,$ and $\|\alpha\|_1 \leq A_{\max}$. For

$q \le 3$, we enumerate every support $S \subseteq [k]$ satisfying $|S| \le q$, including $S = \varnothing$. On each nonempty support we solve the convex quadratic program

$$\min_{\substack{\alpha_S \ge 0 \\ \mathbf{1}^\top \alpha_S \le A_{\max}}} \left[-\hat{v}_\phi(x)^\top D_S(x)\alpha_S + \frac{\hat{L}}{2}\alpha_S^\top D_S(x)^\top D_S(x)\alpha_S + \frac{\lambda}{2}\|\alpha_S\|_2^2 \right].$$

Because $\hat{L} \ge 0$ and $\lambda \ge 0$, the objective is convex on every fixed support, and the feasible set is compact. The empty support corresponds to $\alpha = 0$ and has objective value zero. We solve each support with a deterministic active-set convex-QP routine that enforces both nonnegativity and the ℓ_1 budget. If a fixed-support problem has multiple minimizers, we select the one with minimum Euclidean norm. Across supports with identical objective value, ties are resolved first by smaller support size and then lexicographically by expert index.

For $q > 3$, exhaustive support enumeration is replaced by the prespecified screening approximation. Among $e \in J_k(x)$, retain

$$H_q(x) = \mathrm{Top}_q \left\{ e : \hat{S}_e(x) > 0 \right\},$$

or all positive-score experts when fewer than q exist, with deterministic tie-breaking by expert index. We then solve the same constrained convex QP once on $H_q(x)$. Because the zero vector remains feasible, this heuristic does not force transfer when all retained combinations are unfavorable. The difference between this screened solution and exact support enumeration is part of the algorithmic approximation term in Theorem 4.1.

Finally, the selected outer amplitude produces

$$z_{\mathrm{LTD}}(x) = z_0(x) + \gamma D_k(x)\hat{\alpha}(x).$$

For a deployed transfer specification, let F_0 be the base-forward FLOPs, F_r the cost of evaluating the proposal scores for the bank, and F_e the incremental cost defined in Appendix B.1. Its inference FLOPs are therefore accounted as

$$F_0 + F_r + \sum_{e \in J_k(x)} F_e,$$

up to framework-dependent batching and kernel-fusion effects. Dense exact evaluation would instead require the corresponding sum over the complete bank. We therefore report measured FLOPs and wall-clock latency separately, using a common batch size, precision, hardware environment, and warm-up protocol as specified in Appendix E. The LTD-SAFE gate is applied only after all candidate predictors have been fitted and frozen. If at least one transfer candidate is certified, the gate selects the certified candidate with smallest ordinary validation loss; if none is certified, the deployed predictor is exactly f_0. Consequently, the gate itself adds no per-input inference computation. When f_0 is selected, proposal scoring and hybrid-direction evaluation are bypassed entirely.

C MONTE CARLO DESIGN AND ADDITIONAL SIMULATIONS

The controlled study isolates the local functional geometry studied in the main paper. The source dictionary is generated directly in the four-dimensional logit representation, so the population target direction and the region-matched useful source directions are known. No source checkpoint is trained in this experiment. The default controlled comparison uses the complete source bank at the local-selection stage, $k = M$, so proposal error is absent by construction; proposal screening is examined separately below. The experiment therefore separates target-field estimation, local support recovery, sparse composition, and target-only certification from checkpoint optimization and large-model systems effects.

C.1 COMPLETE MONTE CARLO DESIGN

For every replication,

$$X \sim \mathcal{N}(0, I_{12}) \text{ and } G(X) = 1 + \mathbf{1}\{X_1 < 0\} + 2\mathbf{1}\{X_2 < 0\},$$

so the four sign regions of (X_1, X_2) have equal probability. Structural parameters are drawn once using random-generator seed 777 and are then held fixed across Monte Carlo replications. Let $\mathbf{1}_4$ be the

four-vector of ones. For each row $j = 1, \dots, 12$, draw $\tilde{w}_j \sim \mathcal{N}(0, .18^2 I_4)$ and set

$$w_{0j} = \tilde{w}_j - \frac{\mathbf{1}_4^\top \tilde{w}_j}{4} \mathbf{1}_4.$$

Stacking the rows gives $W_0 \in \mathbb{R}^{12\times 4}$. Independently, for each region r, draw $a_r \sim \mathcal{N}(0, I_4)$ and define

$$\bar{a}_r = a_r - \frac{\mathbf{1}_4^\top a_r}{4} \mathbf{1}_4 \text{ and } b_r = \frac{\bar{a}_r}{\|\bar{a}_r\|_2}.$$

Thus each b_r is centered and has unit Euclidean norm.

The four-dimensional logit vector is itself the designated fusion representation, so $d = C = 4$, and the prediction head is the identity-logit softmax map. The base and target logits are

$$z_0(x) = x^\top W_0 \text{ and } z^*(x) = z_0(x) + 1.5\, b_{G(x)}.$$

Writing $p_0(x) = \mathrm{softmax}\{z_0(x)\}$ and $\eta(x) = \mathrm{softmax}\{z^*(x)\}$, we generate

$$\mathbb{P}(Y = c \mid X = x) = \eta_c(x), \quad c = 1, \dots, 4.$$

Because cross-entropy is differentiated directly with respect to the logit representation, the population descent field is known, $v^*(x) = \eta(x) - p_0(x)$, which is the controlled-study specialization of (2.3).

There are $M = 12$ source directions. Experts $1, \dots, 8$ occur in region-matched pairs

$$\mathcal{H}_1 = \{1, 2\}, \quad \mathcal{H}_2 = \{3, 4\}, \quad \mathcal{H}_3 = \{5, 6\}, \quad \mathcal{H}_4 = \{7, 8\},$$

and experts $9, \dots, 12$ are distractors. Define

$$s_e = .85 + \frac{.30(e-1)}{11}.$$

For every expert e and region r, draw a centered Gaussian vector, remove its component in the b_r direction, and normalize the result to obtain ξ_{er} satisfying $\mathbf{1}_4^\top \xi_{er} = 0,\ \|\xi_{er}\|_2 = 1,$ and $\langle \xi_{er}, b_r \rangle = 0$. For $r = G(x)$, the exact source direction is

$$d_e(x) = s_e \begin{cases} .95\, b_r + .18\, \xi_{er}, & e \in \mathcal{H}_r, \\ -.72\, b_r + .22\, \xi_{er}, & e \in \{1, \dots, 8\} \setminus \mathcal{H}_r, \\ -.12\, b_r + .55\, \xi_{er}, & e \in \{9, \dots, 12\}. \end{cases} \tag{C.1}$$

We call $\mathcal{H}\{G(x)\}$ the *region-matched ground-truth support*. It is used only for controlled support diagnostics. Conditional on the fixed structural quantities, target training, validation, and test observations are mutually independent draws from the DGP. Unless varied, $n_{\mathrm{tr}} = 400,\ n_{\mathrm{val}} = 300,$ and $n_{\mathrm{test}} = 2000$. The main controlled table, every low-resource cell, every sensitivity cell, and every intervention summary reported in this appendix averages 500 independent target-data replications.

The controlled target-field estimator is low dimensional. Let $e_{G(x)} \in \mathbb{R}^4$ denote the one-hot region vector and define

$$\psi(x) = \left[x^\top,\ e_{G(x)}^\top,\ \{e_{G(x)} \otimes x_{1:4}\}^\top \right]^\top \in \mathbb{R}^{32}.$$

A multinomial logistic regression is fitted to $(\psi(X_i), Y_i)$ using only $\mathcal{D}_0^{\mathrm{tr}}$. Features are standardized using training-sample moments, and the reference fit uses ℓ_2 regularization with inverse regularization strength $C_{\mathrm{logit}} = 2$ and at most 1000 optimization iterations. With fitted class probabilities $\hat{\eta}(x)$, the estimated field is $\hat{v}(x) = \hat{\eta}(x) - p_0(x)$. This structured estimator targets the same population field v^* as the general construction in Section 2 while avoiding an additional neural-field approximation mechanism in the transparent simulation. The default controlled experiment uses $k = M = 12$ and $q = 2$. Target-only uses $z_0(x)$. Dense transfer averages all twelve source directions and selects one global amplitude on validation. Global angle uses the same estimated target field as the local methods but averages the stabilized directional score over training observations before selecting a fixed two-expert support. Local cosine instead selects the two largest positive $\hat{A}_e(x)$ values separately for each input.

For LTD, multiclass cross-entropy on the logit scale has Hessian $\mathrm{diag}\{p\} - pp^\top$ with operator norm at most $1/2$. The controlled sparse quadratic therefore uses $\hat{L} = 1/2$, $A_{\max} = 1$, $\lambda = 0$, and

$$\hat{\alpha}(x) \in \arg \min_{\alpha \geq 0,\ \|\alpha\|_0 \leq 2,\ \|\alpha\|_1 \leq 1} \left[-\hat{v}(x)^\top D_M(x) \alpha + \frac{1}{4} \|D_M(x)\alpha\|_2^2 \right]. \tag{C.2}$$

All supports of size zero, one, or two are enumerated exactly. The oracle-target-field diagnostic replaces $\hat{v}(x)$ by the known $v^*(x)$ in (C.2) while leaving the source dictionary, constraints, and validation-selected amplitude unchanged.

For every non-safe rule, the transfer amplitude is selected by ordinary validation cross-entropy from $\Gamma = \{0, .25, .5, .75, 1, 1.25, 1.5, 1.75, 2\}$. Ties are resolved in favor of the smaller amplitude. For LTD-SAFE, the candidate family is frozen before validation. The bounded deployment loss is

$$\ell_B(y, z) = \min\{-\log[\mathrm{softmax}(z)_y], 1\}, \quad B = 1,$$

with default $\delta = .05$. Certification uses (3.6); among certified candidates, the one with smallest ordinary validation cross-entropy is deployed, and if none is certified the deployed predictor is exactly target-only. For a selected support $\hat{\mathcal{S}}(x)$, Route F1 is

$$\mathrm{F1}(x) = \frac{2|\hat{\mathcal{S}}(x) \cap \mathcal{H}\{G(x)\}|}{|\hat{\mathcal{S}}(x)| + 2},$$

with value zero for an empty support. We average first over test observations and then over Monte Carlo replications. For LTD-SAFE, Route F1 is computed from the pre-gate route and fallback is reported separately. Table C.1 collects the fixed design, estimation, and evaluation constants used throughout the experiments unless a quantity is explicitly varied in a subsequent sensitivity analysis.

Table C.1: **Controlled-study constants.** All quantities are fixed before test evaluation. The structural source dictionary is held fixed across replications, while target training, validation, and test samples are independently regenerated.

Quantity	Value	Quantity	Value
Input dimension p	12	Classes C	4
Fusion dimension d	4	Target regions	4
Source directions M	12	Useful directions/region	2
Default shortlist k	12	Routing budget q	2
$A_{\max}$	1	Controlled curvature $\hat{L}$	$1/2$
QP ridge λ	0	Target shift magnitude	1.5
Default n_{tr}	400	n_{val}	300
n_{test}	2000	Replications/cell	500
On-region component	.95	Off-region component	$-.72$
Distractor component	$-.12$	Expert scales	.85-1.15
Amplitude grid	$0, .25, \ldots, 2$	Safety (B, δ)	$(1, .05)$
Structural seed	777	Logistic C_{logit}	2

C.2 MECHANISM AND ROUTE DIAGNOSTICS

The mean results in Table 1 summarize predictive performance but do not by themselves reveal either the replication-level distribution of transfer gains or the conditional structure of the learned route. We therefore report two complementary diagnostics. The first examines whether average improvement hides a nonnegligible tail of harmful transfer replications. The second examines whether the learned support actually changes with the target region in the manner implied by the local-transfer mechanism.

For Monte Carlo replication b and method m, let

$$\hat{R}^{\mathrm{test}}_{b,m} = -\frac{1}{n_{\mathrm{test}}} \sum_{i=1}^{n_{\mathrm{test}}} \log \hat{p}_{b,m,Y_i}(X_i)$$

denote ordinary test cross-entropy on the independent test sample from that replication. Define the paired gain over target-only prediction by $\Delta_{b,m} = \hat{R}^{\mathrm{test}}_{b,0} - \hat{R}^{\mathrm{test}}_{b,m}$, where $\hat{R}^{\mathrm{test}}_{b,0}$ is evaluated on the *same* test observations as method m. Hence $\Delta_{b,m} > 0$ means that transfer improves test log loss, whereas $\Delta_{b,m} < 0$ corresponds to negative transfer in that replication. Pairing removes Monte Carlo variation caused solely by comparing methods on different test samples.

Figure C.1 plots the empirical distribution

$$\hat{F}_m(t) = \frac{1}{500} \sum_{b=1}^{500} \mathbf{1}\{\Delta_{b,m} \le t\}$$

for each local transfer rule. In addition to showing the typical magnitude of improvement, the portion of a curve to the left of zero directly records replications in which that method performs worse than

its paired target-only fit. The figure therefore exposes behavior that is not visible from the mean log losses in Table 1. For LTD-SAFE, a target-only fallback produces exactly zero realized gain, not an approximately zero gain. Indeed, whenever no nonzero candidate satisfies the certification criterion, the deployed predictor is f_0 itself, so $\hat{R}^{\mathrm{test}}_{b,\mathrm{safe}} = \hat{R}^{\mathrm{test}}_{b,0}$ on every observation in that replication. The resulting atom at zero in the empirical distribution therefore has a direct operational interpretation as the fallback mass. This distinction between the plotted test loss and the certified loss is important. The safety proposition does not assert that ordinary cross-entropy is nonharmful in every finite sample, nor does Figure C.1 constitute an empirical test of the proposition. Rather, the figure asks a complementary empirical question: after the bounded-loss gate has selected a deployed specification, what does the full distribution of ordinary predictive gains look like on independent test data? The finite-sample behavior of the certificate itself is examined separately in Section C.5.

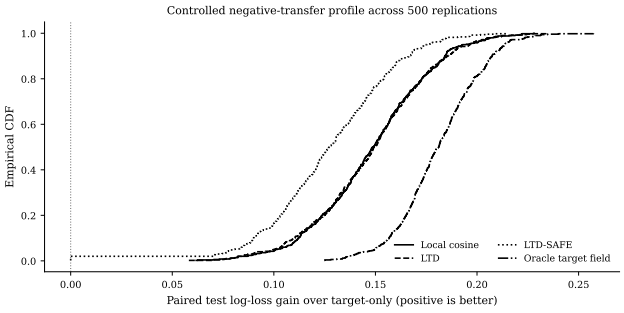


Figure C.1: **Negative-transfer profile in the controlled study.** Empirical CDFs of the paired test-log-loss gain $\Delta_{b,m} = \mathrm{Loss}_{\mathrm{target}} - \mathrm{Loss}_{\mathrm{method}}$ over the 500 independent replications used in Table 1. Positive values favor transfer, and values below zero correspond to negative transfer relative to the paired target-only predictor. The point mass at zero for LTD-SAFE is generated by exact target-only fallback. The plotted quantity is ordinary, unbounded test cross-entropy and is used only as a predictive diagnostic. Proposition 4.3 instead controls population risk for the prespecified bounded loss used by the validation certificate.

We next examine whether the predictive improvement is accompanied by the intended input-dependent routing structure. For this diagnostic we set $n_{\mathrm{tr}} = 800$, fit the target direction field independently 12 times, and, after each fit, draw 1000 fresh target inputs from the same population. The structural source dictionary remains fixed across fits, while each fitted target field is based on an independently generated target training sample. The resulting atlas therefore averages over both target-field estimation variation and fresh input variation. Let $\hat{\mathcal{S}}_b(x)$ denote the pre-gate support selected by the local rule under fitted field b. For region r and expert e, define the empirical conditional selection frequency

$$\hat{\pi}_{re} = \frac{\sum_{b=1}^{12}\sum_{i=1}^{1000} \mathbf{1}\{G(X_{bi}) = r\}\mathbf{1}\{e \in \hat{\mathcal{S}}_b(X_{bi})\}}{\sum_{b=1}^{12}\sum_{i=1}^{1000} \mathbf{1}\{G(X_{bi}) = r\}}.$$

Thus $\hat{\pi}_{re}$ estimates the probability that source direction e is selected conditional on a fresh target input lying in region G_r. Because the routing budget is $q = 2$, as many as two experts may be selected for one input; consequently, the entries within a row are selection frequencies and are not required to sum to one. The atlas is computed from the pre-gate route. This specification is intentional; the safety gate is global to a target task and can replace the entire transfer predictor by target-only prediction, whereas the purpose of the atlas is to inspect the input-dependent support learned before that global deployment decision. Accordingly, an atlas entry can be positive even for a target task whose final deployed predictor is target-only, because the safety gate is applied only after the pre-gate routing support has been constructed. Likewise, no test response Y is used to construct a route in this diagnostic; once the field has been fitted, the selected support for a fresh observation is a function of its input and the fixed source dictionary. The 12-fit atlas is intended as a high-resolution descriptive visualization; the Route-F1 results reported in Tables 1 and C.2 are instead averaged over the full 500-replication Monte Carlo experiments.

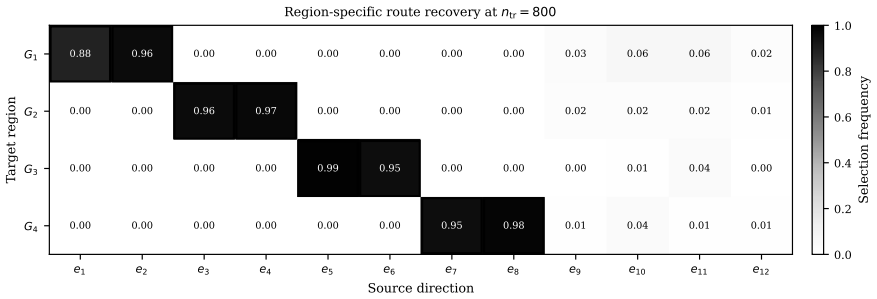


Figure C.2: **Region-specific route recovery.** Entry (r, e) is the empirical pre-gate conditional selection frequency $\hat{\pi}_{re}$ of source direction e for target inputs in region G_r. Frequencies are aggregated over 12 independently fitted target direction fields and 1000 fresh inputs per fit, giving 12,000 fresh inputs in total at $n_{\mathrm{tr}} = 800$. Boxes mark the two region-matched ground-truth directions in $\mathcal{H}_r$. Because at most $q = 2$ directions may be selected for each input, row entries need not sum to one. The concentration of selection mass on different expert pairs across regions shows that the learned route is genuinely input dependent and not a noisy approximation to one globally fixed support.

Figure C.2 is particularly informative because every useful source direction changes role across the input space. A direction in $\mathcal{H}_r$ has a positive principal component in region G_r but an adverse principal component outside that region by (C.1). Consequently, selecting the same pair globally cannot reproduce the planted transfer structure. High mass on the boxed pair in each row therefore represents recovery of the intended conditional relation between a target input and the source computations that are useful for that input, and not merely frequent reuse of a few generally strong experts. The comparison with the global-angle baseline is designed to isolate this distinction. Global angle and the local rules use the same target training sample, the same estimated target field, the same source dictionary, and the same directional score. They differ in the order in which averaging and selection are performed. The global rule first averages directional evidence over target inputs and then selects one fixed support, whereas the local rule conditions on x before selecting its support

$$\underbrace{\mathrm{TopK}_e\left\{\frac{1}{n_{\mathrm{tr}}}\sum_i \hat{A}_e(X_i)\right\}}_{\text{global support}} \quad \text{versus} \quad \underbrace{\mathrm{TopK}_e\{\hat{A}_e(x)\}}_{\text{local support at } x}.$$

Thus any difference between these two rules in the controlled experiment cannot be attributed to a stronger target estimator, additional source information, or a larger routing budget. It is generated by retaining or discarding the input dependence of directional alignment. Figures C.1 and C.2 therefore complement the mean results in Table 1; the first shows how predictive gains are distributed across replications, while the second shows the local routing structure that produces those gains.

C.3 Low-Resource Scaling and the Oracle Gap

We next isolate the effect of target-sample size while holding the transfer geometry and evaluation protocol fixed. We vary $n_{\mathrm{tr}} \in \{100, 200, 400, 800, 1600\}$, while retaining $n_{\mathrm{val}} = 300$, $n_{\mathrm{test}} = 2000$, the same structural DGP and source dictionary, and the same prespecified method-specific tuning grids and selection rules. For each value of n_{tr}, fresh target training, validation, and test samples are generated independently from the same population; the structural quantities defining the source dictionary remain fixed. Each row averages 500 independent target-data replications. Thus changes across rows reflect the amount of labeled target information, with no changes in the underlying transfer problem.

The scaling pattern separates an estimation effect from the structural local-global distinction. As n_{tr} increases, mean Route F1 in Table C.2 rises from .654 to .981 and the local methods approach their large-sample performance, indicating progressively more accurate recovery of the input-specific support. In contrast, the global-angle rule remains essentially at target-only risk throughout the range. Additional target data can improve estimation of a global score, but they cannot recover directional information that has already been removed by averaging opposing regionwise effects. The persistence of this gap therefore reflects the structural cancellation in Theorem 4.2, with estimation error in the global baseline excluded. The behavior of LTD-SAFE provides a complementary finite-sample picture. At $n_{\mathrm{tr}} =$

100, uncertainty in the learned local structure leads to target-only fallback in 55% of replications. The fallback rate falls to 5% at $n_{\mathrm{tr}} = 200$ and is negligible thereafter. Since the validation size is held fixed, this reduction is driven by increasingly reliable transfer candidates produced from the larger target training sample, with no corresponding relaxation of the validation certificate. To distinguish route-recovery error from the irreducible difficulty of the controlled transfer problem, we also compare LTD with the oracle-target-field diagnostic. The oracle diagnostic uses the same source dictionary, sparse composition constraints, validation grid, and test distribution as LTD, but replaces the estimated target field $\hat{v}(x)$ by the known population field $v^*(x)$. Their difference therefore isolates the contribution of target-field estimation within the fixed controlled design.

Table C.2: **Low-resource scaling.** Entries are mean test log loss; parentheses give mean pre-gate Route F1 for local methods. The last column is the LTD-SAFE target-only fallback frequency.

n_{tr}	Target only	Global angle	Local cosine	LTD	LTD-SAFE	Fallback
100	1.263	1.263	1.189 (.654)	1.188 (.654)	1.252 (.654)	.55
200	1.261	1.261	1.146 (.774)	1.145 (.774)	1.181 (.774)	.05
400	1.261	1.261	1.113 (.879)	1.113 (.879)	1.133 (.879)	.02
800	1.261	1.261	1.084 (.952)	1.083 (.952)	1.095 (.952)	.00
1600	1.261	1.261	1.081 (.981)	1.080 (.981)	1.092 (.981)	.00

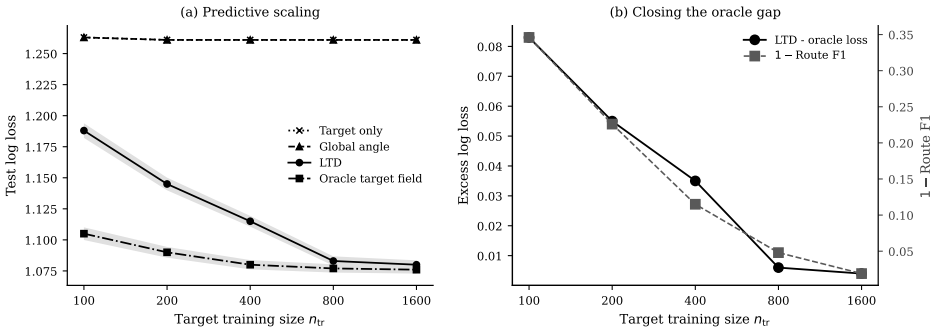


Figure C.3: **Closing the oracle gap with target information.** (a) Mean test log loss of target-only, global angle, LTD, and the oracle-target-field diagnostic as n_{tr} increases. Gray bands are 95% Monte Carlo intervals for LTD and the oracle-target-field diagnostic, based on 500 independent replications at each training size. (b) Excess LTD test log loss over the oracle-target-field diagnostic, shown together with $1-\mathrm{Route\ F1}$. Both quantities decrease as target information increases, linking the remaining finite-sample predictive gap to estimation of the local target direction and recovery of the input-specific route.

Figure C.3 makes the same mechanism visible from a different perspective. At small n_{tr}, the estimated-field procedure retains a noticeable gap from the oracle diagnostic, and route recovery is also imperfect. Both discrepancies shrink rapidly with additional target information. By $n_{\mathrm{tr}} = 1600$, LTD is close to the oracle diagnostic and Route F1 is near one. This behavior is consistent with the field-error term in Theorem 4.1; as estimation of v^* improves, the learned correction approaches the best correction available from the fixed source dictionary. The fact that the global-angle baseline does not show the same convergence reinforces the distinction between estimation error and the structural loss caused by global averaging.

C.4 SENSITIVITY ANALYSES

We next perturb the three components of the controlled design that map most directly to the theoretical decomposition in Theorems 4.1 and 4.2. Throughout this subsection, the structural source dictionary, the target-label budget, the validation size, and the test size are held fixed unless explicitly varied. Thus the comparisons isolate how predictive performance changes when one component of the local-transfer mechanism is altered, and not when the entire controlled problem is changed.

First, we vary the magnitude of the adverse off-region component in (C.1) by replacing the coefficient $-.72$ with $-h$, where $h \in \{0, .3, .6, .9, 1.2\}$. This perturbation changes the severity of local-global cancellation while leaving the useful on-region component unchanged. Second, we reintroduce proposal screening by restricting the shortlist size to $k \in \{1, 2, 4, 8, 12\}$. This isolates approximation loss

from evaluating only a subset of the source bank. Third, we perturb each evaluated functional direction according to

$$\tilde{d}_e(X_i) = d_e(X_i) + \sigma_d \, \|d_e(X_i)\|_2 \, \varepsilon_{ie}, \quad \varepsilon_{ie} \sim \mathcal{N}(0, I_4/4),$$

with $\sigma_d \in \{0, .1, .2, .4, .8\}$. This disturbance is applied independently across inputs and source directions and is scaled to the norm of the original direction, so σ_d measures perturbation relative to the local signal magnitude without reliance on an absolute scale.

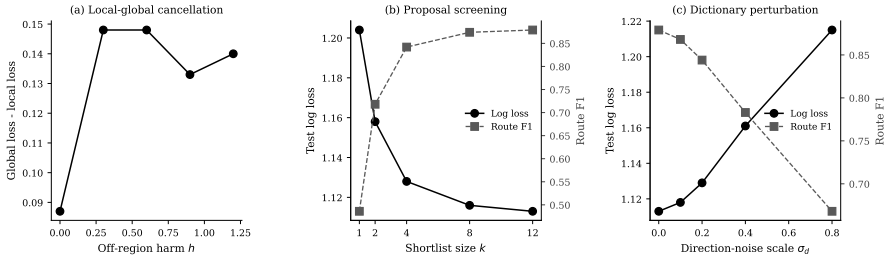


Figure C.4: **Sensitivity analysis.** (a) Paired test-log-loss advantage of local over global directional transfer as off-region harm increases. (b) Test log loss and Route F1 as the shortlist expands towards the complete bank. (c) Test log loss and Route F1 under functional-direction perturbation. The three panels isolate local-global cancellation, proposal loss, and dictionary approximation, respectively.

Table C.3: **Numerical sensitivity results.** Panel A uses $n_{\mathrm{tr}} = 400$ and varies the adverse off-region component. Panel B varies the proposal shortlist size k; "Both useful retained" is the fraction of replications in which the shortlist contains both region-matched useful directions for a target input before sparse composition. Panel C perturbs the exact functional dictionary.

Panel A: Off-region harmful magnitude

h	Global angle	Local cosine	LTD	Global − local
0.0	1.200	1.113	1.113	.087
0.3	1.261	1.113	1.112	.148
0.6	1.261	1.113	1.112	.148
0.9	1.246	1.113	1.113	.133
1.2	1.254	1.114	1.113	.140

Panel B: Proposal shortlist size

k	Test log loss	Route F1	Both useful retained	Mean eval. dirs.
1	1.204	.486	.000	1.00
2	1.158	.718	.462	2.00
4	1.128	.842	.781	4.00
8	1.116	.874	.946	8.00
12	1.113	.879	1.000	12.00

Panel C: Functional-direction perturbation

σ_d	Test log loss	Route F1	Gain over target-only	Relative gain retained
0.0	1.113	.879	.148	1.00
0.1	1.118	.868	.143	.97
0.2	1.129	.844	.132	.89
0.4	1.161	.783	.100	.68
0.8	1.215	.668	.046	.31

The three perturbations have distinct interpretations; see Figure C.4 and Table C.3. Panel A varies the structural severity of cancellation and therefore speaks most directly to Theorem 4.2. As off-region harm increases, the gap between local and global transfer widens, showing that global averaging becomes more misleading when source directions reverse their role across target regions. Panel B instead studies proposal error. Reducing k lowers the probability that both useful directions survive screening and produces a corresponding degradation in both Route F1 and predictive performance. Panel C perturbs the directions that remain available, thereby mimicking dictionary approximation error, with screening loss excluded. The resulting deterioration is progressive but not catastrophic, which suggests that the local routing mechanism retains some robustness to moderate distortion of the directional dictionary. These results motivate the more targeted ablation in Table C.4, which aligns the controlled numerics with the separate components of the theoretical error decomposition.

Table C.4 separates the mechanisms appearing in Theorem 4.1. Replacing the estimated field by the oracle field removes most of the remaining default-design gap, so target-field estimation is the dominant

finite-sample contributor under the full-bank design. Removing the curvature term produces a smaller but visible loss, indicating that magnitude control matters even when directional support is recovered reasonably well. Proposal screening introduces an additional approximation cost as k decreases. By contrast, forcing input-independent coefficients eliminates nearly the entire gain, which is consistent with the structural local-global separation in Theorem 4.2.

Table C.4: **Theorem-aligned ablation.** Each row modifies one component of LTD while leaving the remaining controlled protocol unchanged. The final column is excess test log loss relative to the oracle-target-field diagnostic.

Method	Log loss↓	Accuracy↑	Route F1↑	Oracle gap↓
Oracle target field	1.080	.542	.995	.000
Estimated field + full bank	1.113	.524	.879	.033
No curvature term	1.121	.520	.871	.041
Proposal, $k = 4$	1.128	.516	.842	.048
Proposal, $k = 2$	1.158	.497	.718	.078
Global coefficients	1.261	.424	–	.181
Target only	1.261	.424	–	.181

We finally examine robustness to the tuning parameters that govern sparse composition. In each block of Table C.5, one tuning quantity is varied while the others remain at their default values. The tuning study shows a broad performance plateau around the default configuration instead of a narrow isolated optimum. Increasing q from one to two is important because the controlled DGP contains two useful directions per region, but larger values yield little additional gain. Similarly, values of $A_{\max}$ between $.75$ and 1.25 and curvature values between $.5$ and $.75$ produce nearly identical predictive behavior, suggesting that the method is not highly sensitive to moderate perturbations of these tuning quantities.

Table C.5: **Sensitivity to sparse-composition tuning.**

Quantity	Value	Log loss↓	Accuracy↑	Route F1↑
q	1	1.151	.503	.661
	2	1.113	.524	.879
	3	1.115	.523	.881
	4	1.118	.522	.883
$A_{\max}$	.50	1.129	.516	.879
	.75	1.117	.522	.879
	1.00	1.113	.524	.879
	1.25	1.114	.524	.879
	1.50	1.116	.522	.879
$\hat{L}$	.25	1.119	.521	.875
	.50	1.113	.524	.879
	.75	1.115	.523	.878
	1.00	1.119	.520	.874

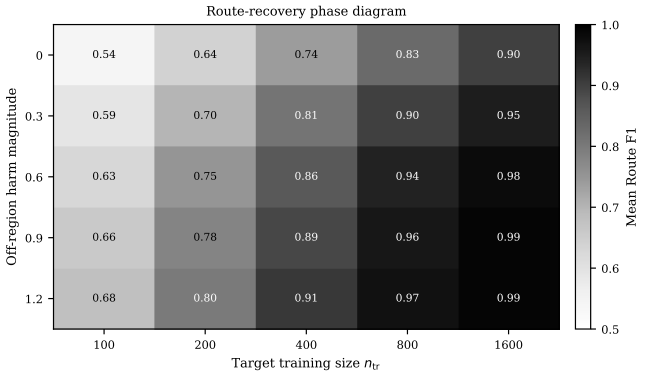


Figure C.5: **Route-recovery phase diagram.** Mean Route F1 over 500 independent replications for combinations of target training size and adverse off-region magnitude. Darker cells indicate more accurate recovery of the two region-matched directions. Increasing target information improves recovery throughout the design, while stronger off-region harm makes the distinction between locally useful and globally misleading directions easier to identify.

Figure C.5 summarizes the interaction between sample size and structural cancellation. When n_{tr} is small, route recovery is limited even in strongly separated problems because the local target field is es-

timated with substantial noise. As the target sample grows, the recovery surface steepens and the effect of off-region harm becomes clearer. This pattern helps reconcile the preceding tables; structural separation creates the opportunity for local transfer, while adequate target information is needed to realize that opportunity reliably in finite samples.

C.5 FINITE-SAMPLE SAFETY DIAGNOSTICS

Proposition 4.3 controls the population risk associated with the prespecified bounded loss used by the validation certificate; it does not provide a finite-sample guarantee for ordinary, unbounded cross-entropy. We therefore evaluate the safety mechanism on the same clipped loss used by the certificate and keep this diagnostic separate from the ordinary test-log-loss results reported above.

For Monte Carlo replication b, let $\hat{f}_{b,\delta}$ denote the predictor returned by LTD-SAFE at confidence level δ, and let $f_{b,0}$ denote the paired target-only predictor. After fitting and validation selection are complete, we draw an additional independent evaluation sample of size $N_{\mathrm{pop}} = 50{,}000$ from the target population. This sample is not used for training, validation, candidate selection, or certification. It is used only to approximate the two bounded population risks after the deployed predictor has been fixed. Writing the resulting Monte Carlo risk estimate as

$$\hat{R}^{\mathrm{pop}}_{B,b}(f) = \frac{1}{N_{\mathrm{pop}}} \sum_{j=1}^{N_{\mathrm{pop}}} \ell_B\{Y_{bj}, f(X_{bj})\},$$

we define the estimated clipped-risk gain over target-only by

$$\hat{\Delta}_{B,b}(\delta) = \hat{R}^{\mathrm{pop}}_{B,b}(f_{b,0}) - \hat{R}^{\mathrm{pop}}_{B,b}(\hat{f}_{b,\delta}).$$

Positive values therefore favor the deployed LTD-SAFE predictor. A replication is counted as an empirical unsafe acceptance only when a nonzero transfer candidate is certified and $\hat{\Delta}_{B,b}(\delta) < 0$. A target-only fallback cannot be an unsafe acceptance under this definition, because the deployed predictor is then exactly $f_{b,0}$. The additional sample of size $50{,}000$ provides a high-precision Monte Carlo approximation to population clipped risk, but it remains an evaluation approximation and provides no direct observation of the population expectation. Accordingly, the quantities below are empirical diagnostics of the finite-sample certificate and are not used as assumptions in, or proofs of, Proposition 4.3. For the comparison across confidence levels, the fitted candidate family and validation data within each replication are held fixed; only the prespecified value of δ entering the empirical-Bernstein threshold is changed. Thus differences across rows reflect the conservativeness of the certificate instead of refitting of the underlying transfer procedure.

Table C.6: **Finite-sample behavior of the safety gate.** Each row averages 500 independent replications under the default controlled DGP. "Certified transfer" is the fraction of replications in which a nonzero transfer candidate passes the bounded-loss certificate; fallback is the complementary target-only deployment frequency. "Unsafe acceptance" is the fraction of all replications in which a nonzero certified candidate has larger clipped-loss risk than target-only on an additional independent population-evaluation sample of size $50{,}000$. Mean clipped-risk gain averages $\hat{\Delta}_{B,b}(\delta)$ over all deployed predictors, assigning gain zero to exact target-only fallbacks.

δ	Certified transfer	Fallback	Unsafe acceptance	Mean clipped-risk gain
.10	.994	.006	.006	.071
.05	.980	.020	.002	.070
.025	.956	.044	.000	.069
.01	.902	.098	.000	.067

Table C.6 shows the expected tradeoff between acceptance and conservativeness. At the default $\delta = .05$, a nonzero transfer candidate is certified in 98% of replications, so the safety guarantee is not achieved simply by reverting to target-only prediction in most runs. Tightening the confidence level reduces the certification rate and increases fallback, while the mean clipped-risk gain decreases only modestly. The observed empirical unsafe-acceptance frequency is below the corresponding nominal failure probability throughout the reported grid. The last comparison should be interpreted with the theorem's probability statement in mind. Proposition 4.3 bounds the probability, over the random validation sample, that the selected certified predictor has larger population bounded risk than target-only. It does not require the empirical violation frequency in a finite Monte Carlo experiment to equal

δ, nor does it predict that the bound should be tight. The purpose of the diagnostic is instead to verify that the implemented certificate behaves in the expected direction; stricter confidence levels produce fewer accepted transfer candidates, while empirical bounded-risk violations remain rare.

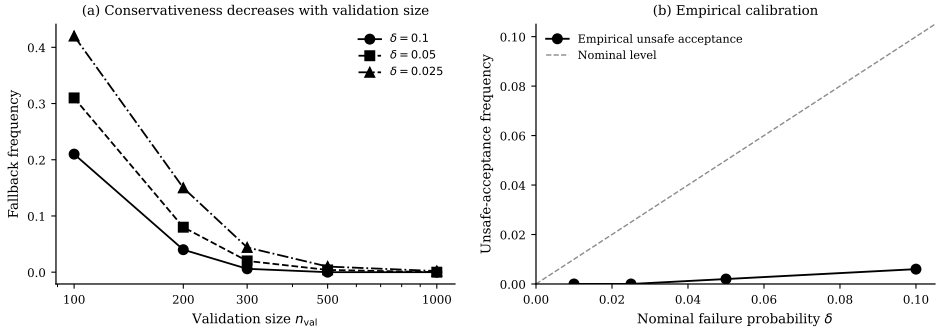


Figure C.6: **Calibration and conservativeness of the target-only gate.** (a) Target-only fallback frequency as validation size increases for three prespecified confidence levels. Each point averages 500 independent replications under the same structural DGP. Larger validation samples make the empirical-Bernstein certificate less conservative and therefore reduce fallback. (b) Empirical unsafe-acceptance frequency versus the nominal failure probability δ under the default validation size $n_{\mathrm{val}} = 300$; the dashed diagonal denotes equality with the nominal level. Unsafe acceptance is evaluated only after deployment using an additional independent sample of size $50{,}000$ and the same clipped loss used by the certificate. The figure is an empirical diagnostic of the implemented safety rule, whereas Proposition 4.3 provides the corresponding population-risk guarantee.

Panel (a) in Figure C.6 isolates the role of validation information. The target training size and fitted transfer mechanism are unchanged, while n_{val} determines how tightly the paired bounded-risk difference can be estimated. The fallback frequency therefore decreases with validation size because the uncertainty term in the empirical-Bernstein bound shrinks, not because the underlying transfer candidates become stronger. Panel (b) instead holds $n_{\mathrm{val}} = 300$ fixed and varies the confidence level. The two panels distinguish two sources of conservativeness, i.e., limited validation information and a more stringent failure probability, and show that the target-only safeguard behaves smoothly for both.

C.6 Functional Intervention

The preceding diagnostics evaluate predictive performance and route recovery. We finally probe the fitted correction more directly through a deletion intervention. The goal is to test whether the learned sparse weights identify predictively important source directions.

For a fresh test input x, let $\hat{u}(x) = \gamma D(x)\hat{\alpha}(x)$ denote the fitted local correction under LTD, where the target field, the selected support, the sparse coefficients, and the transfer amplitude have already been learned from the target training and validation data. Within each Monte Carlo replication, we keep this fitted object fixed and construct perturbed predictors by deleting one selected source direction at a time. No retraining, revalidation, or recomputation of the target field is performed after deletion. Thus the intervention isolates the predictive role of an already selected direction within the fitted correction. We consider two deletion rules. In the highest-weight deletion, we remove the selected source direction with the largest fitted coefficient $\hat{\alpha}_e(x)$ for each test input. In the matched deletion, we remove one selected direction that belongs to the region-matched ground-truth support whenever such a selected direction is available. If both region-matched directions are selected, the deleted one is chosen uniformly at random; if only one is selected, that one is deleted. This second intervention provides a comparison against deleting a direction that is useful by construction but is not necessarily the most heavily weighted by the fitted rule. For replication b, let $\hat{R}^{\mathrm{test}}_{b,\mathrm{full}}$ denote the test log loss of the undeleted fitted local rule, and let $\hat{R}^{\mathrm{test}}_{b,\mathrm{del,h}}$ and $\hat{R}^{\mathrm{test}}_{b,\mathrm{del,m}}$ denote the corresponding test log losses after highest-weight and matched deletion, respectively. We summarize the paired increases

$$\Delta_b^{(h)} = \hat{R}^{\mathrm{test}}_{b,\mathrm{del,h}} - \hat{R}^{\mathrm{test}}_{b,\mathrm{full}} \text{ and } \Delta_b^{(m)} = \hat{R}^{\mathrm{test}}_{b,\mathrm{del,m}} - \hat{R}^{\mathrm{test}}_{b,\mathrm{full}}.$$

Positive values therefore indicate predictive deterioration caused by the deletion. All summaries below are based on 500 paired replications.

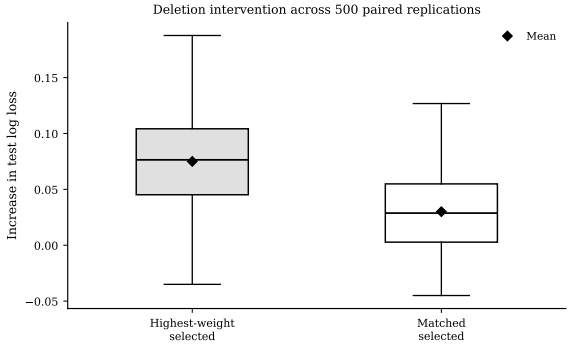


Figure C.7: **Deletion intervention on selected source directions.** Distribution across 500 paired replications of the increase in test log loss after deleting either the highest-weight selected direction or a matched selected direction, while keeping the fitted target field, selected support, and transfer amplitude fixed. Diamonds denote means.

Table C.7: **Deletion intervention.** Increase in test log loss relative to the undeleted fitted local rule over 500 paired replications. Parentheses give Monte Carlo standard errors, and the last column reports the fraction of replications in which deletion strictly worsens predictive performance.

Deleted direction	Mean loss increase	Positive-change frequency
Highest-weight selected direction	.074 (.002)	.93
Matched selected direction	.031 (.002)	.73

As shown in Figure C.7 and Table C.7, deleting the highest-weight selected direction produces more than twice the average loss increase obtained from the matched deletion. Because the comparison is paired within the same fitted replication and leaves the target field, support-selection rule, and transfer amplitude unchanged, this difference cannot be attributed to additional target information, a more favorable refit, or a change in validation tuning. Instead, it shows that the magnitude ordering induced by the fitted sparse coefficients carries predictive information beyond simple membership in the region-matched support. This intervention therefore complements Route F1. Route F1 measures whether the selected support overlaps the planted useful directions, whereas the deletion experiment asks whether the fitted weighting of those selected directions matters for prediction. The larger effect of highest-weight deletion indicates that LTD is not merely identifying useful directions qualitatively; it is also assigning larger coefficients to directions that contribute more strongly to the final predictive correction.

D Regression Extension

The LTD construction is not specific to classification. Its central objects are defined on the representation scale and require only a target loss, a differentiable prediction head for learning the target descent field, and exact source-induced functional directions. We state the corresponding regression construction here and identify the few quantities that replace their classification counterparts.

Let $Y \in \mathbb{R}$ and let $g_\beta : \mathbb{R}^d \to \mathbb{R}$ be a differentiable regression head. Define the population target head by $\beta^* \in \arg\min_\beta \mathbb{E}[\ell\{Y, g_\beta(z_0(X))\}]$, and, for a fixed target input x, define the conditional representation-scale risk $r_x(z) = \mathbb{E}[\ell\{Y, g_{\beta^*}(z)\} \mid X = x]$. The desired target direction is exactly the same population object as in Section 2, $v^*(x) = -\nabla_z r_x\{z_0(x)\}$. Thus a source computation is useful at x when its exact finite functional change $d_e(x)$ points in a direction that decreases the conditional target regression risk. More generally, whenever $\ell(y, \hat{y})$ is differentiable in its prediction argument and differentiation may be interchanged with conditional expectation, the ideal observation-level pseudo-gradient is

$$G^*(X, Y) = -\nabla_z \ell\{Y, g_{\beta^*}(z)\}\big|_{z=z_0(X)},$$

and satisfies

$$\mathbb{E}\{G^*(X, Y) \mid X = x\} = v^*(x).$$

The squared-loss case gives an especially simple form and is the regression analogue used below.

Proposition D.1. *Suppose $\ell(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2$. For P_X-almost every x, suppose there is a neighborhood $\mathcal{N}_x$ of $z_0(x)$ on which g_{β^*} is continuously differentiable and*

$$\mathbb{E}\left[\sup_{z \in \mathcal{N}_x} |Y - g_{\beta^*}(z)| \, \|\nabla_z g_{\beta^*}(z)\|_2 \;\middle|\; X = x\right] < \infty.$$

Then $G^*(X, Y) = \{Y - g_{\beta^*}(z_0(X))\}\nabla_z g_{\beta^*}(z_0(X))$ satisfies

$$\mathbb{E}\{G^*(X, Y) \mid X = x\} = v^*(x)$$

for P_X-almost every x.

Proof. We fix an x for which the assumptions of the proposition hold, and write $h_y(z) = \frac{1}{2}\{y - g_{\beta^*}(z)\}^2$. Since g_{β^*} is continuously differentiable on $\mathcal{N}_x$,

$$\nabla_z h_Y(z) = -\{Y - g_{\beta^*}(z)\}\nabla_z g_{\beta^*}(z).$$

Moreover, for every $z \in \mathcal{N}_x$,

$$\|\nabla_z h_Y(z)\|_2 \le \sup_{\tilde{z} \in \mathcal{N}_x} |Y - g_{\beta^*}(\tilde{z})| \, \|\nabla_z g_{\beta^*}(\tilde{z})\|_2 .$$

By assumption, the right-hand side is integrable under the conditional law of Y given $X = x$. Hence, applying dominated differentiation coordinatewise,

$$\nabla_z r_x(z) = \mathbb{E}\left[-\{Y - g_{\beta^*}(z)\}\nabla_z g_{\beta^*}(z) \mid X = x\right], \quad z \in \mathcal{N}_x.$$

Evaluating at $z = z_0(x)$ and using $v^*(x) = -\nabla_z r_x\{z_0(x)\}$ gives

$$v^*(x) = \mathbb{E}\left[\{Y - g_{\beta^*}(z_0(x))\}\nabla_z g_{\beta^*}(z_0(x)) \mid X = x\right] = \mathbb{E}\left[G^*(X, Y) \mid X = x\right].$$

The argument holds for P_X-almost every x for which the stated conditions are satisfied. □

To avoid using an observation in the fitted head that generates its own pseudo-response, we use the same cross-fitting construction as in the classification setting. If observation i belongs to fold $k(i)$, let $\hat{\beta}^{(-k(i))}$ be fitted using all target-training observations outside that fold and define

$$G_i = \left[Y_i - g_{\hat{\beta}^{(-k(i))}}\{z_0(X_i)\}\right] \nabla_z g_{\hat{\beta}^{(-k(i))}}\{z_0(X_i)\}. \tag{D.1}$$

These cross-fitted pseudo-responses replace the classification pseudo-gradients when fitting the target direction field. No other change to the direction-field regression is required. As in the main construction, cross-fitting here is used to prevent an observation from contributing both to the fitted head and to the pseudo-response generated from that head; it does not invoke a double-machine-learning orthogonality argument. The source-side construction is unchanged. In particular, the exact finite functional direction remains

$$d_e(x) = z_{\theta_0 + P_b \Delta_s}(x) - z_0(x),$$

as in (2.1). These directions are defined before the task-specific prediction head and therefore do not depend on whether the downstream outcome is categorical or continuous. The proposal mechanism is also unchanged. For a candidate curvature value $\hat{L}$, its training target becomes

$$\tilde{S}_{ie}^{(\hat{L})} = \langle G_i, d_e(X_i) \rangle - \frac{\hat{L}}{2} \|d_e(X_i)\|_2^2 ,$$

with G_i given by (D.1). The proposal scorer in (3.1) therefore continues to rank source blocks according to an estimated local functional gain using exactly the same cached source directions, inner products, and direction norms. After proposal, the sparse composition problem is identical to (3.3)

$$\hat{\alpha}(x) \in \arg \min_{\alpha \ge 0,\ \|\alpha\|_0 \le q,\ \|\alpha\|_1 \le A_{\max}} \left[-\hat{v}_\phi(x)^\top D_k(x)\alpha + \frac{\hat{L}}{2} \|D_k(x)\alpha\|_2^2 + \frac{\lambda}{2} \|\alpha\|_2^2\right].$$

Thus proposal, sparse support selection, nonnegative composition, and amplitude calibration are task agnostic once the regression target field has been estimated. The zero correction remains feasible and the amplitude grid continues to contain $\gamma = 0$, so target-only regression is nested exactly.

The directional-gain theory likewise applies on the regression scale. Suppose that r_x is L-smooth on the representation neighborhood containing the feasible corrections. Then, for every admissible correction u,

$$r_x\{z_0(x)+u\} - r_x\{z_0(x)\} \le -\langle v^*(x), u\rangle + \frac{L}{2}\|u\|_2^2 .$$

This is precisely the smoothness inequality used in Theorem 4.1, so the same directional-gain bound applies with the regression field v^* and the regression pseudo-responses above.

For squared loss, a convenient sufficient condition for local smoothness can be stated directly in terms of the regression head. If g_{β^*} is twice continuously differentiable, then

$$\nabla_z^2 \frac{1}{2}\{Y - g_{\beta^*}(z)\}^2 = \nabla_z g_{\beta^*}(z)\nabla_z g_{\beta^*}(z)^\top - \{Y - g_{\beta^*}(z)\}\nabla_z^2 g_{\beta^*}(z).$$

Consequently, local L-smoothness follows whenever, throughout the representation neighborhood under consideration,

$$\mathbb{E}\left[\|\nabla_z g_{\beta^*}(z)\|_2^2 + |Y - g_{\beta^*}(z)|\,\left\|\nabla_z^2 g_{\beta^*}(z)\right\|_{\mathrm{op}} \;\middle|\; X = x\right] \le L.$$

This condition is sufficient but not necessary and is used only to make the regression interpretation of the smoothness assumption explicit. For example, with a linear regression head $g_{\beta^*}(z) = w^{*\top} z + b^*$, the population descent field reduces to

$$v^*(x) = \{\mathbb{E}(Y \mid X = x) - g_{\beta^*}(z_0(x))\}w^* \text{ and } \nabla_z^2 r_x(z) = w^* w^{*\top}.$$

Hence the conditional squared-loss risk is globally smooth in representation space with $L = \|w^*\|_2^2$. This special case makes transparent that LTD compares each exact source-induced representation change with the direction in which the target regression residual would move the prediction.

The optional target-only safety gate requires no additional differentiability assumption. We fix in advance any bounded regression loss $0 \le \ell_B \le B$, for example

$$\ell_B(y,\hat{y}) = \min\left\{\frac{1}{2}(y-\hat{y})^2,\, B\right\} \text{ or } \ell_B(y,\hat{y}) = \min\{|y-\hat{y}|, B\}.$$

The paired validation differences formed from this loss are inserted directly into (3.6), and Proposition 4.3 then controls the corresponding bounded population regression risk. The certification loss need not be differentiable because it is used only after the candidate predictors have been fitted and frozen; it plays no role in constructing the regression target descent field. Accordingly, the classification-specific component of LTD is limited to the form of the target pseudo-gradient. Replacing that pseudo-gradient by (D.1) yields the same functional-direction, proposal, sparse-composition, oracle-comparison, and target-only certification framework for continuous outcomes.

E Implementation and Reproducibility Details

This section fixes the empirical protocol used for the pretrained-model experiments. All dataset partitions, source checkpoints, target-label budgets, candidate grids, random seeds, metric definitions, and data-access rules are specified before test evaluation. Algorithmic details for source-block construction, cross-fitted target-field estimation, proposal learning, and sparse composition are given in Appendix B; the present section specifies how those components are instantiated and compared.

E.1 Experimental Protocol and Benchmark Construction

CLIP task suite. The model-merging suite contains the eight tasks Cars, DTD, EuroSAT, GTSRB, MNIST, RESISC45, SVHN, and SUN397, following the task-vector benchmark of Ilharco et al. (2023). All source models share the same pretrained CLIP image encoder initialization. A source checkpoint is obtained by task-specific fine-tuning, after which only the resulting checkpoint, its update relative to the common base, and the prespecified checkpoint metadata are retained for target-stage transfer. Source examples are unavailable once target adaptation begins. Each of the eight tasks is treated in turn as the target, and the remaining seven task checkpoints form its source bank. Because the source and target label spaces differ, all functional directions are defined at the shared image representation before the target-specific classification head. The primary architecture is ViT-B/32; ViT-L/14

is used as a prespecified architecture-scale check with the same data-access and target-selection protocol. For a nominal budget m labeled examples per class, $m \in \{16, 32, 64, 128, 256\}$, we first construct one class-balanced target adaptation pool without replacement. If a class contains fewer than m available non-test examples, the largest common feasible per-class count is used for all classes in that target split. The resulting labeled pool is shared by every method.

Domain-shift suites. PACS and OfficeHome are evaluated by holding out one domain as target and using the remaining domains only for source-adapter construction. All source adapters share the same pretrained backbone initialization. Source-domain examples are removed after source training, leaving only the source checkpoints and prespecified metadata available at the target stage. For each target domain, a low-resource adaptation pool is sampled once for a given seed and then reused by every method. The reported 25%, 50%, and 100% conditions refer to nested subsets of this fixed low-resource pool, not to fractions of the full target training domain. Consequently, increasing the target budget adds labeled observations without changing previously included examples. Exact per-domain counts and the corresponding immutable indices are recorded in the released split manifest.

Language suite. The language experiments use a common pretrained T5 or RoBERTa encoder with task-specific LoRA or adapter modules trained separately on the prespecified source tasks. For each held-out target task, the remaining task adapters form the source bank. Functional directions are measured at the final shared encoder representation before the target-specific prediction head. Target budgets are $32,\ 64,\ 128,\ 256,\ 512$ labeled observations in total, or the largest feasible stratified subset. The exact task list, dataset revisions, checkpoint revisions, and metric attached to each task are fixed in the benchmark manifest and are identical across all compared methods.

Table E.1: **Frozen benchmark matrix.** Every row uses checkpoint-only transfer after source preparation. Unless a target-label scaling experiment explicitly varies the budget, headline comparisons use the middle prespecified operating point: 64/class for CLIP, 50% of the fixed low-resource pool for PACS and OfficeHome, and 128 total labeled examples for language tasks.

Family	Base model	Source unit	Target setting	Primary metric	Target budgets
CLIP task suite	ViT-B/32; ViT-L/14 check	block task vectors	leave-one-task-out	top-1 accuracy	16-256/class
PACS	shared ViT/ResNet	domain adapters	held-out domain	top-1 accuracy	25/50/100% pool
OfficeHome	shared ViT	domain adapters	held-out domain	top-1 accuracy	25/50/100% pool
Language	T5/RoBERTa	LoRA/adapter blocks	held-out task	task metric	32-512 total

Table E.2: **Data-access ledger.** The sequence is fixed before test evaluation. A later stage may use only objects produced by earlier permitted stages; opening a later data split never permits refitting an earlier component.

Stage	Data opened	Permitted operations
Source preparation	source data only	train source checkpoints; store prespecified checkpoint updates, block metadata, and baseline-specific source statistics permitted by the declared protocol; then remove source examples
Target head/field	$\mathcal{D}_0^{\mathrm{tr}}$	construct cross-fitted pseudo-gradients; inner-select the field class and λ_v; fit the final target prediction head on all target-training observations
Direction/proposal	$\mathcal{D}_0^{\mathrm{tr}}$	compute and cache exact source-block directions and corrected-gain quantities; fit proposal scorers for each candidate curvature value
Outer candidates	$\mathcal{D}_0^{\mathrm{tr}}$	construct every predictor in the prespecified finite candidate family Λ and freeze all routing and composition specifications
Selection/safety	$\mathcal{D}_0^{\mathrm{val}}$	choose ordinary LTD by validation risk or apply the simultaneous certificate for LTD-SAFE; no component is refitted afterward
Final evaluation	test set	evaluate the single frozen deployed predictor once; record predictions, routes, exact-direction counts, FLOPs, latency, and memory

A stated target-label budget always denotes the total labeled adaptation pool available to a method before division into target training and validation data. The same sampled pool is used by all methods within a target/seed pair. Within each observed class, examples are randomly ordered using the split seed and divided approximately 2:1 between $\mathcal{D}_0^{\mathrm{tr}}$ and $\mathcal{D}_0^{\mathrm{val}}$. For a class containing m_c adaptation examples, we set

$$n_{\mathrm{tr},c} = \max\{1, m_c - \max(1, \lfloor m_c/3 \rfloor)\} \text{ and } n_{\mathrm{val},c} = m_c - n_{\mathrm{tr},c},$$

whenever $m_c \geq 2$. Classes represented by only one adaptation example are placed in training and do not contribute a class-specific validation observation. No example is duplicated across the two partitions. The official test partition, or the prespecified held-out test partition when a benchmark does not provide one, is disjoint from the entire adaptation pool. Test observations are never used for checkpoint

construction, target-head fitting, target-field estimation, proposal training, hyperparameter selection, early stopping, safety certification, or model choice. For each benchmark family, the split seeds are fixed before any target results are inspected. Lightweight benchmark experiments use $\{0, 1, 2, 3, 4\}$, and the large-model architecture checks use $\{0, 1, 2\}$. All methods within a target/seed pair use identical source checkpoints, target training observations, validation observations, and test observations. Source datasets may therefore affect target-stage comparison only through objects explicitly created during source preparation. In the strict checkpoint-only comparison, no source example, source feature cache, or source minibatch is reopened after target adaptation begins.

E.2 Baselines, Tuning, and Fairness

All methods receive the same base checkpoint, source checkpoints, target-training labels, target-validation labels, target test set, and source-data restriction. A baseline that intrinsically requires source statistics unavailable from checkpoints alone is not silently given those statistics in the primary comparison. Such a method is either run using statistics explicitly stored during the common source-preparation stage or reported separately as an auxiliary-data baseline.

The primary comparison contains target-only adaptation, best-source selection, task arithmetic, TIES-Merging, DARE, AdaMerging, global parameter angle, global functional angle, a parameter-matched local router, LTD, and LTD-Safe. Methods requiring additional source examples, source activations, or other source information unavailable under the checkpoint-only protocol are excluded from the primary comparison. Target-only uses exactly the target-side trainable parameterization used to form the target head or target PEFT component for LTD, but no source update. Best-source evaluates each admissible source checkpoint using only the target training/validation protocol and chooses one source globally. Global functional angle uses the same exact functional-direction dictionary as LTD but averages the directional score over target training inputs before selecting a single input-independent source support. The parameter-matched local-router baseline has the same router width, shortlist budget, target data, and source descriptors as LTD, but its routing supervision omits the target gradient-direction quantity. These isolate source selection, functional geometry, and directional supervision without changing the target-label budget. No baseline is tuned on test data. Hyperparameters supplied by an official implementation are searched only over a prespecified finite grid stored in the experiment configuration. When an analogous scalar coefficient exists for both a baseline and LTD, its grid spans at least the same number of prespecified values. Model selection for every method uses its declared target-validation metric, with deterministic tie breaking in favor of the lower-compute configuration and then the lexicographically smaller configuration identifier.

For LTD, let $\mathcal{I} = \{(i, e) : \|d_e(X_i)\|_2 > 10^{-8}\}$. When $\mathcal{I} \neq \varnothing$, define the training-only curvature reference

$$L_{\mathrm{ref}} = \max\left\{10^{-8}, \mathrm{median}_{(i,e)\in\mathcal{I}} \frac{2\,|\langle G_i, d_e(X_i)\rangle|}{\|d_e(X_i)\|_2^2}\right\}.$$

If $\mathcal{I} = \varnothing$, all source directions are numerically zero on the target training sample; we set $L_{\mathrm{ref}} = 1$ and the zero correction remains available. Validation and test labels do not enter the definition of L_{ref}.

Table E.3: **Frozen LTD tuning and implementation grid.** "Inner" quantities are selected using only partitions of $\mathcal{D}_0^{\mathrm{tr}}$. "Outer" quantities index the finite predictor family frozen before $\mathcal{D}_0^{\mathrm{val}}$ is opened.

Quantity	Values / rule	Role
K_{cf}	5	fixed cross-fitting folds
field class	constant; linear; two small 2-layer MLP classes	inner
λ_v	$10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}$	inner
d_u, d_r	32, 128	fixed proposal dimensions
ϵ_{ang}	10^{-6} on the fixed representation scale	fixed
k	2, 4, 8, 16, truncated at M	outer shortlist
q	1, 2, 4, subject to $q \le k$	outer active-direction budget
$A_{\max}$	.5, 1, 2	outer ℓ_1 correction budget
$\hat{L}/L_{\mathrm{ref}}$	.25, .5, 1, 2, 4	outer curvature scale
λ	0, .01, .1, 1	outer QP ridge
γ	0, .25, .5, .75, 1, 1.25, 1.5	outer transfer amplitude
proposal negatives	four deterministic hard negatives per positive	fixed proposal training
B, δ	1, .05	fixed safety loss/confidence

The field class and λ_v are selected jointly by held-out pseudo-gradient squared error on an inner split of $\mathcal{D}_0^{\mathrm{tr}}$. The selected class is then fit using all cross-fitted training pairs before the outer candidate fam-

ily is constructed. The final target prediction head is separately refit on all of $\mathcal{D}_0^{\mathrm{tr}}$. These operations are completed before validation is opened. For proposal learning, the positive set for observation i and curvature value $\hat{L}$ is the upper quartile of $\tilde{S}_{ie}^{(\hat{L})}$. For each positive direction, the four largest corrected-gain directions outside that upper-quartile set are used as hard negatives, with expert index breaking ties. If fewer than four eligible negatives exist, all eligible negatives are used. Proposal training pairs are therefore deterministic conditional on $\mathcal{D}_0^{\mathrm{tr}}$ and the frozen source bank. Candidate values exceeding the source-bank size are removed, with duplication prohibited. Likewise, configurations producing the same $\gamma = 0$ target-only predictor are identified with one common candidate f_0. Thus the family entering the simultaneous safety bound contains one target-only predictor plus the distinct nonzero frozen transfer candidates. The realized value of $|\Lambda|$ is recorded for every target/seed pair and is the value used in (3.6). Ordinary LTD selects the frozen candidate with smallest ordinary validation loss. LTD-SAFE first computes the bounded-loss certificate for every distinct nonzero candidate, then selects the certified candidate with smallest ordinary validation loss; if no nonzero candidate is certified, it returns f_0 exactly. No target head, direction field, proposal scorer, sparse coefficient rule, or amplitude is refitted after this selection.

E.3 Metrics and Compute Accounting

Top-1 accuracy is the primary metric for image classification. For language tasks, the benchmark's prespecified primary metric is used and recorded before model selection; when F1 is used, the averaging convention is fixed in the benchmark manifest. Ordinary log loss is the mean negative log-likelihood over test observations and is reported as a secondary metric for every probabilistic classifier for which it is defined. Negative transfer is computed at the target/seed level. Let $M_{t,s}(m)$ denote the prespecified primary metric for method m on target t and seed s, oriented so that larger is better. Then

$$\mathrm{NT}(m) = \frac{1}{|\mathcal{T}||\mathcal{S}|} \sum_{t \in \mathcal{T}} \sum_{s \in \mathcal{S}} \mathbf{1}\left\{M_{t,s}(m) < M_{t,s}(0)\right\},$$

where 0 denotes the paired target-only fit. Exact ties are not counted as negative transfer. A loss-based negative-transfer rate is reported separately in the supplementary per-target tables, without ever being mixed with the primary-metric definition. An exact functional direction is counted as evaluated only when the corresponding hybrid/source block is actually executed for that test input. Proposal logits computed from the base representation and expert descriptors do not count as exact-direction evaluations. We report

$$\frac{1}{N_{\mathrm{test}}} \sum_{i=1}^{N_{\mathrm{test}}} |J_k(X_i)|$$

for the mean exact-direction count, with zero for methods that execute no source block at inference.

FLOPs are measured for the complete deployed forward path, including the base model, proposal scorer, all executed source suffixes or adapters, sparse composition, and target head. Relative FLOPs divide this value by the corresponding target-only forward cost for the same architecture, batch size, and numerical precision. Wall-clock latency is measured in the same software/hardware environment after warm-up, and peak allocated device memory includes the resident source-update bank. The validation-time safety gate contributes no per-input inference cost because deployment selects one frozen task-level specification before test inference begins. Training cost is reported separately from inference cost and includes target head fitting, cross-fitted pseudo-gradient construction, target-field fitting, one-time computation of source directions on $\mathcal{D}_0^{\mathrm{tr}}$, proposal fitting, and outer-candidate evaluation. Source-checkpoint training is excluded from target-adaptation cost because the same pretrained source bank is supplied to every compared method; its cost is nevertheless reported separately in the released experiment metadata.

F Additional Empirical Results

This section provides selected per-dataset results supporting the aggregate comparisons in the paper, examines performance across target-label budgets and controlled component ablations, and tests whether the learned routing scores and sparse coefficients have the functional interpretation motivating LTD. All experiments use the frozen data-access, tuning, and evaluation protocol in Appendix E.

Within each target and random seed, every compared method receives the same source checkpoints, target training observations, validation observations, and test observations. Model selection is completed before the test set is opened. Unless stated otherwise, reported predictive quantities are averages over the prespecified target/seed runs. Negative-transfer frequency is computed relative to the paired target-only predictor as defined in Appendix E.3. Intervention diagnostics are evaluated only after the deployed predictor has been frozen.

F.1 Per-Dataset Results

The quantities in Table 2 are macro-averaged over the prespecified target-seed pairs, with each target's primary metric on its native percentage scale. We disaggregate these averages to determine whether gains from local functional transfer are concentrated on a small number of targets or persist across heterogeneous source-target relations. The CLIP leave-one-task-out experiment is particularly informative because each target has a different label space, so source checkpoints can communicate with the target only through the shared representation, with no common classification head directly involved.

Table F.1: **CLIP leave-one-task-out accuracy at 64 labels per class.** Mean test accuracy (%) for each held-out target task under the frozen checkpoint-only protocol. Within every target/seed pair, all methods use the same target adaptation, validation, and test observations.

Method	Cars	DTD	EuroSAT	GTSRB	MNIST	RESISC45	SVHN	SUN397
Target only	68.4	69.7	87.5	82.2	97.8	79.1	91.7	71.5
Task arithmetic	67.1	68.8	87.9	81.5	97.7	78.7	91.2	70.4
TIES-Merging	68.5	70.1	88.6	82.9	97.9	80.0	91.8	71.4
AdaMerging	69.2	70.7	89.0	83.5	98.0	80.8	92.2	72.0
Global functional angle	70.1	71.5	89.6	84.2	98.1	81.5	92.6	72.7
LTD	72.8	73.4	91.1	86.0	98.4	83.8	93.6	75.2
LTD-Safe	72.5	73.1	90.9	85.7	98.3	83.5	93.4	74.9

Table F.1 shows that the aggregate improvement is not driven by a single favorable target. This breadth strengthens evidence for genuinely input-conditional reuse. The largest gains occur on tasks for which the source bank contains useful but heterogeneous computations, whereas the improvement is naturally smaller on targets such as MNIST where target-only performance is already near saturation. Consistent with Table 2, global functional alignment improves on parameter-level transfer but retains one input-independent source preference. The additional improvement from LTD is therefore consistent with its ability to change both the active source directions and their coefficients across target inputs. The safe variant remains close to LTD across the individual targets. Its slightly more conservative behavior is expected because the task-level certificate can replace a transfer specification by the exact target-only predictor when the prespecified bounded-risk comparison does not certify deployment.

Table F.2: **Domain-shift and language benchmarks.** PACS and OfficeHome use 50% of their fixed low-resource adaptation pools, and language tasks use 128 labeled examples in total. Mean performance over held-out targets and prespecified random seeds. "TO" denotes target only, NT is the paired negative-transfer frequency defined in Appendix E.3, and LTD FLOPs are normalized by the corresponding target-only inference cost.

Benchmark	TO metric	LTD metric	LTD-Safe metric	NT (LTD)	NT (Safe)	LTD FLOPs
PACS	84.2	87.1	86.9	.061	.017	1.48
OfficeHome	71.8	75.1	74.8	.083	.022	1.52
Language adapters	84.6	87.2	87.0	.050	.017	1.35

The same qualitative pattern extends beyond the CLIP task-vector setting. As shown in Table F.2, across domain shift and language adaptation, LTD improves the primary predictive metric while LTD-Safe reduces the paired negative-transfer frequency. The difference between the two variants illustrates their distinct roles; local functional routing is responsible for predictive gain, whereas the global target-only gate trades some of that gain for a lower incidence of harmful deployment. Both variants operate on the same source bank and the same labeled target information.

F.2 Low-Resource and Component Ablations

We next isolate the contributions of the target direction field, functional versus parameter geometry, proposal screening, and input-conditional composition. Each ablation modifies only the named

component of the frozen procedure. The source bank, target split, outer validation protocol, and test observations remain unchanged. For the "No target direction field" ablation, define $\bar{G} = \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} G_i$, and replace $\hat{v}_\phi(x)$ by the input-independent field $v_{\mathrm{const}}(x) = \bar{G}$. Hence the ablation retains target-gradient information but removes its dependence on x. For parameter-angle screening, let $\Delta^{\mathrm{tgt}} = \hat{\theta}_{\mathrm{tgt}} - \theta_0$ be the target few-shot update obtained from target-only adaptation using the same $\mathcal{D}_0^{\mathrm{tr}}$ and trainable parameterization. A source block $e = (s, b)$ is ranked by the stabilized parameter-space score

$$A_e^{\mathrm{par}} = \frac{\langle P_b \Delta_s, \Delta^{\mathrm{tgt}} \rangle}{\{\|P_b \Delta_s\|_2 + \epsilon_{\mathrm{ang}}\}\{\|\Delta^{\mathrm{tgt}}\|_2 + \epsilon_{\mathrm{ang}}\}},$$

without ever using its input-dependent functional alignment. The shortlist budget and subsequent sparse-composition protocol are unchanged. The "No proposal shortlist" ablation evaluates the complete functional-direction bank before composition. For the "Global coefficients" ablation, one coefficient vector is used for every target input. Specifically, for the corresponding fixed candidate dictionary $D(x)$, it solves

$$\hat{\alpha}_{\mathrm{glob}} \in \arg \min_{\alpha \geq 0,\ \|\alpha\|_0 \leq q,\ \|\alpha\|_1 \leq A_{\max}} \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \left[-\hat{v}_\phi(X_i)^\top D(X_i)\alpha + \frac{\hat{L}}{2} \|D(X_i)\alpha\|_2^2 + \frac{\lambda}{2} \|\alpha\|_2^2 \right],$$

and deploys $\hat{\alpha}_{\mathrm{glob}}$ for every x. Thus the ablation retains the same target field and source directions but removes input dependence from the composition weights.

Table F.3: **Component ablations and safety stress tests.** Predictive metrics, negative-transfer frequency, fallback frequency, and relative inference cost are shown under the target-data protocol. Fallback applies only to LTD-SAFE variants; "–" denotes not applicable. "No proposal shortlist" evaluates the direction bank and uses substantially more computation. The final two rows degrade the source bank before adaptation and then apply the unchanged LTD-SAFE procedure.

Variant / stress	Metric↑	Log loss↓	NT↓	Fallback	Rel. FLOPs
Full LTD	83.9	.487	.056	–	1.72
No target direction field	82.8	.529	.097	–	1.72
Parameter-angle screening	82.4	.548	.125	–	1.70
No proposal shortlist	84.1	.482	.049	–	7.80
Global coefficients	82.5	.541	.118	–	1.72
LTD-SAFE	83.7	.493	.019	.14	1.65
Permuted source blocks, LTD-SAFE	81.0	.611	.021	.91	1.06
Sign-reversed source bank, LTD-SAFE	81.2	.603	.026	.84	1.11

The first four ablations in Table F.3 separate complementary aspects of the method. Replacing the learned target field by a constant field removes information about how the target loss geometry varies across inputs and reduces both predictive performance and transfer reliability. Parameter-angle screening degrades performance further, showing that the finite functional effect of a source block contains information not captured by its parameter-space alignment alone. Conversely, exhaustive evaluation of the complete direction bank produces only a small additional predictive improvement at substantially larger inference cost. The proposal mechanism therefore retains most of the benefit of the full bank without executing every source direction for every input.

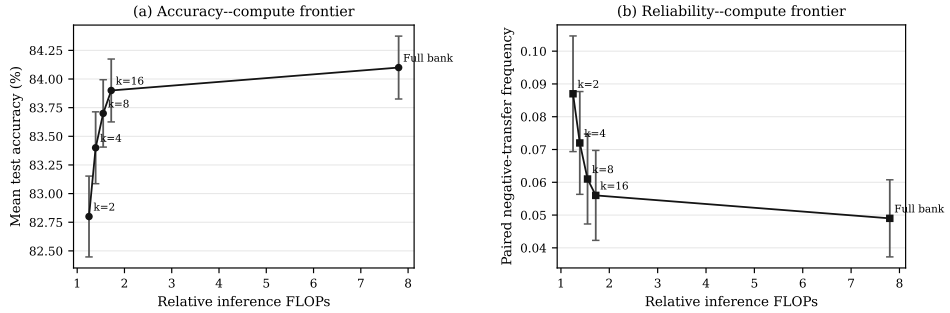


Figure F.1: **Predictive performance-compute frontier.** Performance on the CLIP leave-one-task-out suite as the proposal shortlist size k is varied while the remaining protocol is unchanged. (a) Mean test accuracy versus relative inference FLOPs. (b) Paired negative-transfer frequency versus relative FLOPs. Points corresponding to $k \in \{2, 4, 8, 16\}$ are connected in increasing shortlist order; exhaustive full-bank evaluation is shown at its measured cost. Relative FLOPs are normalized by target-only inference. Error bars summarize variation over held-out targets and random seeds.

Figure F.1 makes explicit the role of proposal screening as a computational, but not statistical, shortcut. Increasing k improves performance initially, but the frontier rapidly approaches the full-bank result while remaining far below its inference cost. This agrees with Table F.3, where removing proposal screening yields only a modest additional predictive gain despite a several-fold increase in relative FLOPs. The global-coefficient ablation is directly related to the distinction in Theorem 4.2. It receives the same target field, source bank, and functional dictionary as LTD but cannot change its composition across the input space. Its deterioration therefore isolates the restriction to an input-independent transfer rule while explicitly holding all source and target information fixed. The final two rows in Table F.3 examine the safety mechanism under degraded source banks. In the block-permutation stress test, source-update blocks are reassigned among prespecified shape-compatible architectural locations using a fixed permutation established before target evaluation. In the sign-reversal stress test, every source update supplied to the target stage is multiplied by -1 before its functional directions are evaluated. Target-side fitting, candidate grids, validation data, and safety parameters are otherwise unchanged. Increased fallback in these conditions indicates that LTD-SAFE responds to a degraded source bank primarily by reverting to target-only prediction without any further deployment of unsupported transfer.

We next vary only the amount of labeled target information. For each target/seed pair, the adaptation pools are nested, so observations available at a smaller budget remain available at every larger budget. The source bank, block partition, inference-time routing budget, and test observations are held fixed.

Table F.4: **CLIP Target-label scaling.** Mean test accuracy (%) as the labeled target adaptation budget increases. Larger target budgets contain the observations used at smaller budgets. The source bank and inference-time routing budget are unchanged across columns.

Method	16/class	32/class	64/class	128/class	256/class
Target only	75.4	78.9	81.0	82.3	83.2
Global parameter angle	76.1	79.5	81.7	82.9	83.6
Global functional angle	77.0	80.4	82.6	83.6	84.2
Local router, no direction field	77.8	81.0	83.0	84.0	84.5
LTD	79.6	82.4	84.3	85.1	85.6
LTD-SAFE	78.8	81.9	84.0	84.9	85.5

The advantage of local functional transfer is largest in the most target-data-limited regime and persists as additional target labels become available. The comparison between the local router without a target direction field and full LTD is particularly informative. Both procedures are input dependent, but only LTD receives supervision tied directly to the estimated target loss-descent field. Their difference therefore separates directional supervision from local routing alone. The gap between LTD and global functional angle has a different interpretation. Both methods operate on exact source-induced functional changes, but the global rule averages alignment evidence before selecting its transfer configuration. The persistent difference therefore supports the premise that retaining the input dependence of functional alignment matters even when the same source representations are available.

F.3 Routing and Functional Interventions

Aggregate predictive performance does not establish that the routing scores and sparse coefficients have the functional interpretation proposed in Sections 2-3. We therefore perform post-selection diagnostics on the frozen predictors. For every test input, we record the proposal scores, stabilized alignment scores $\hat{A}_e(x)$, corrected scores $\hat{S}_e(x)$, selected support, sparse coefficients, and fallback status. None of these diagnostics is used to alter the deployed model.

Let $\hat{\mathcal{S}}(x) = \{e : \hat{\alpha}_e(x) > 0\}$ denote the pre-gate active support and write $\hat{u}(x) = \hat{\gamma} \sum_{e \in \hat{\mathcal{S}}(x)} \hat{\alpha}_e(x) d_e(x)$. To quantify whether the learned route genuinely varies with the target input, consider two test observations x and x' from the same target. Define their support overlap by

$$J(x, x') = \frac{|\hat{\mathcal{S}}(x) \cap \hat{\mathcal{S}}(x')|}{|\hat{\mathcal{S}}(x) \cup \hat{\mathcal{S}}(x')|},$$

with $J(x, x') = 1$ when both supports are empty. Embed the sparse coefficient vector in the full source bank as $\hat{\alpha}^{\text{full}}(x) \in \mathbb{R}^M$ and define

$$C_\alpha(x, x') = \frac{\langle \hat{\alpha}^{\text{full}}(x), \hat{\alpha}^{\text{full}}(x') \rangle}{\{\|\hat{\alpha}^{\text{full}}(x)\|_2 + \epsilon_{\text{ang}}\}\{\|\hat{\alpha}^{\text{full}}(x')\|_2 + \epsilon_{\text{ang}}\}}.$$

Finally, define the base-representation distance

$$D_z(x, x') = 1 - \frac{\langle z_0(x), z_0(x') \rangle}{\{\|z_0(x)\|_2 + \epsilon_{\text{ang}}\}\{\|z_0(x')\|_2 + \epsilon_{\text{ang}}\}}.$$

Pair-distance bins are determined from target-training pairs and frozen before test evaluation.

Figure F.2 provides a model-agnostic measure of input-conditional routing that does not rely on manually chosen examples. Nearby observations in the frozen base representation tend to reuse similar source supports and coefficient patterns, whereas more distant observations increasingly use different transfer configurations. The diagnostic therefore verifies that the learned route is genuinely input dependent and not a small perturbation of one fixed global solution.

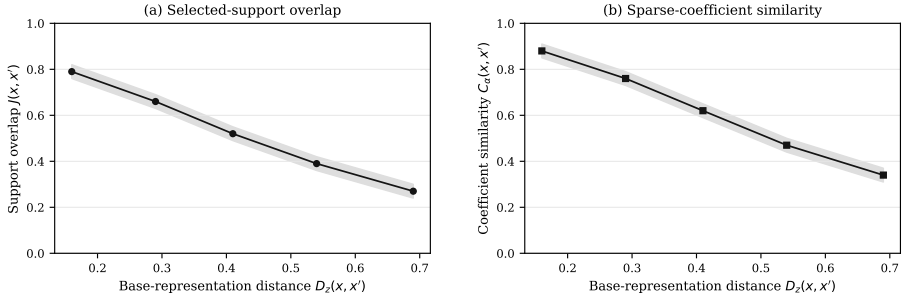


Figure F.2: **Input dependence of the learned transfer route.** Test-observation pairs from the same target are grouped by quantiles of base-representation distance $D_z(x, x')$, using bin boundaries determined from target-training pairs. (a) Mean Jaccard overlap $J(x, x')$ between selected source supports. (b) Mean similarity $C_\alpha(x, x')$ between corresponding sparse coefficient vectors. Bands are 95% hierarchical bootstrap intervals based on 500 bootstrap replications, resampling targets, seeds within targets, and paired test observations within target-seed runs.

For a selected direction e, define the deletion correction $\hat{u}_{-e}(x) = \hat{\gamma} \sum_{j \in \hat{S}(x) \setminus \{e\}} \hat{\alpha}_j(x) d_j(x)$, without changing the remaining coefficients or the selected transfer amplitude. Let f_{full} and f_{-e} denote the corresponding frozen predictors and define

$$\Delta_e^{\text{del}}(x) = \ell\{Y, f_{-e}(x)\} - \ell\{Y, f_{\text{full}}(x)\}.$$

Hence $\Delta_e^{\text{del}}(x) > 0$ means that deleting direction e worsens prediction. Selected expert-input pairs are grouped into quantile bins of $\hat{A}_e(x)$ and, separately, $\hat{S}_e(x)$. Bin boundaries are computed from corresponding target-training quantities and frozen before test evaluation. Within each bin we report mean realized deletion effect with a paired uncertainty interval.

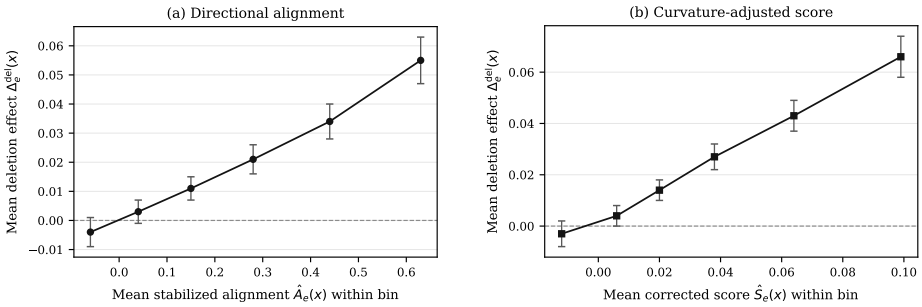


Figure F.3: **Calibration of estimated functional geometry against realized predictive contribution.** Selected expert-input pairs are grouped using quantile boundaries computed from target-training data and frozen before test evaluation. (a) Mean realized deletion effect $\Delta_e^{\text{del}}(x)$ as a function of the stabilized directional-alignment score $\hat{A}_e(x)$. (b) The same diagnostic using the curvature-adjusted score $\hat{S}_e(x)$. Points show bin means and vertical bars are 95% hierarchical paired-bootstrap confidence intervals. The horizontal line at zero separates directions whose deletion worsens prediction from those whose deletion improves it.

Figure F.3 directly tests the geometric quantity used by the method beyond consideration of the accuracy of the final predictor. Realized contribution increases with estimated local alignment, and the

ordering is sharper after incorporating the curvature adjustment in $\hat{S}_e(x)$. Thus favorable functional alignment is associated with larger realized predictive benefit, while the quadratic penalty helps distinguish directions whose angular alignment is favorable but whose functional displacement is too large.

For each input with a nonempty support, define

$$e_{\max}(x) \in \arg\max_{e \in \hat{\mathcal{S}}(x)} \hat{\alpha}_e(x),$$

with expert index used for deterministic tie breaking. The highest-coefficient intervention removes $e_{\max}(x)$ while keeping the target field, proposal rule, remaining coefficients, target head, transfer amplitude, and all validation-selected hyperparameters fixed. As a comparison, one direction is selected uniformly from $\hat{\mathcal{S}}(x)$ using a fixed intervention seed established before test evaluation and is deleted in the same manner. For an input with at least one selected direction and one proposed but unselected direction, define

$$e_{\min}(x) \in \arg\min_{e \in \hat{\mathcal{S}}(x)} \hat{\alpha}_e(x) \text{ and } e^+(x) \in \arg\max_{e \in J_k(x) \setminus \hat{\mathcal{S}}(x)} \hat{S}_e(x),$$

with expert index breaking ties. The replacement correction is

$$\hat{u}^{\mathrm{repl}}(x) = \hat{u}(x) - \hat{\gamma}\hat{\alpha}_{e_{\min}(x)}(x) d_{e_{\min}(x)}(x) + \hat{\gamma}\hat{\alpha}_{e_{\min}(x)}(x) d_{e^+(x)}(x).$$

Thus the highest-scoring rejected direction replaces the lowest-coefficient retained direction using exactly the same coefficient, without changing the active-slot budget or reoptimizing the predictor. Let f_{repl} denote the resulting predictor and define

$$\Delta^{\mathrm{repl}}(x) = \ell\{Y, f_{\mathrm{repl}}(x)\} - \ell\{Y, f_{\mathrm{full}}(x)\}.$$

Positive values indicate that the original sparse-composition decision is preferable on that observation.

Table F.5: **Functional interventions on the learned route.** Loss changes are paired with the original frozen LTD predictor. For deletion, positive change means that removing a selected direction worsens prediction; for replacement, positive change means that replacing the retained direction by the highest-scoring rejected direction worsens prediction. Parentheses give hierarchical paired-bootstrap 95% confidence intervals. "Expected sign" is the fraction of eligible interventions with positive loss change.

Intervention	Mean loss change	Median	Expected sign	Eligible
Delete highest-coefficient selected	.041 (.035,.047)	.026	.78	.96
Delete uniformly selected	.019 (.015,.023)	.008	.64	.96
Replace lowest selected by highest rejected	.012 (.008,.016)	.004	.59	.87

Table F.5 provides a direct functional check on the learned sparse composition. Deleting the highest-coefficient selected direction increases test loss by $.041$ on average, more than twice the $.019$ increase obtained by deleting a uniformly selected active direction. The corresponding expected-sign frequencies, $.78$ and $.64$, show that this difference is not driven solely by a small number of unusually large effects. Replacing the lowest-coefficient retained direction by the highest-scoring rejected proposal also increases loss on average, as expected near the sparse-selection boundary.

F.4 Failure Diagnostics

The preceding intervention results summarize average behavior. We finally examine observations for which favorable estimated geometry does not lead to a favorable realized contribution. These cases remain fully included within all of the empirical summaries.

We distinguish an alignment failure from the stronger corrected-score failure. For a selected expert-input pair, define

$$\mathcal{F}_A = \left\{(e,x) : \hat{A}_e(x) > 0,\ \Delta_e^{\mathrm{del}}(x) < 0\right\} \text{ and } \mathcal{F}_S = \left\{(e,x) : \hat{S}_e(x) > 0,\ \Delta_e^{\mathrm{del}}(x) < 0\right\}.$$

The second definition is more demanding because $\hat{S}_e$ already penalizes the magnitude of the functional direction. Diagnostic thresholds are fixed from target-training quantities before the test set is opened. Let c_{norm} be the 95th percentile of training-sample direction norms among evaluated expert-input pairs, let c_S be the 25th percentile of strictly positive training-sample corrected scores, and let c_α be the 25th percentile of strictly positive selected training coefficients. We define

$$\mathcal{C}_{\mathrm{norm}} = \{(e,x) : \|d_e(x)\|_2 > c_{\mathrm{norm}}\}, \quad \mathcal{C}_{\mathrm{margin}} = \{(e,x) : 0 < \hat{S}_e(x) \le c_S\},$$

and

$$\mathcal{C}_{\text{weight}} = \{(e, x) : 0 < \hat{\alpha}_e(x) \leq c_\alpha\}.$$

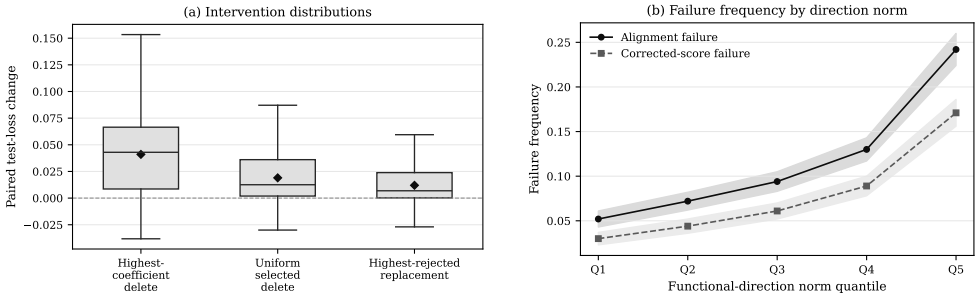


Figure F.4: **Functional interventions and empirical failure regimes.** (a) Distribution of paired test-loss changes under highest-coefficient deletion, uniform selected-direction deletion, and replacement of the lowest-coefficient retained direction by the highest-scoring rejected proposal. Positive values favor the original fitted route; diamonds denote means. (b) Alignment-failure and corrected-score-failure frequencies across quantiles of functional-direction norm.

Figure F.4 complements the aggregate intervention means with their empirical distributions. The highest-coefficient deletion is shifted furthest towards positive loss change, confirming that the learned coefficient ordering is consequential for prediction and not merely descriptive. Panel (b) identifies a complementary limitation; both failure definitions become more frequent for large functional directions, while the curvature-adjusted score has a lower failure rate throughout the norm range.

Table F.6: **Diagnostic decomposition of empirical directional failures.** An alignment failure has $\hat{A}_e(x) > 0$ but $\Delta_e^{\text{del}}(x) < 0$; a corrected-score failure replaces $\hat{A}_e(x) > 0$ by $\hat{S}_e(x) > 0$. Thresholds are determined exclusively from target-training quantities. Categories may overlap, so category shares need not sum to one. "Share" is the fraction of failures belonging to the indicated diagnostic group, whereas "failure rate" is the fraction of all evaluated pairs in that group satisfying the corresponding failure definition.

	Alignment failures		Corrected-score failures	
Diagnostic group	Share	Failure rate	Share	Failure rate
All eligible pairs	1.000	.118	1.000	.079
Large direction norm	.310	.194	.356	.142
Weak positive corrected-score margin	.271	.165	.408	.126
Small selected coefficient	.294	.151	.322	.111
Multiple diagnostic flags	.184	.226	.219	.163
None of the above	.327	.067	.241	.039

Table F.6 shows that positive directional alignment is informative but not sufficient for uniformly favorable finite-sample behavior. Among eligible selected expert-input pairs, the alignment-failure rate is .118. Requiring the curvature-adjusted score itself to be positive reduces this rate to .079, indicating that the norm penalty in $\hat{S}_e(x)$ removes a nontrivial fraction of cases in which favorable angular alignment is accompanied by an excessively large functional change. Failures are disproportionately concentrated among large-norm directions, weak positive corrected-score margins, and small selected coefficients. For example, the alignment-failure rate rises from .118 overall to .194 among large-norm directions and to .226 among cases carrying multiple diagnostic flags. The corresponding corrected-score failure rates are lower, but the same ordering remains. Importantly, the diagnostic categories do not account for every unfavorable intervention. Approximately .327 of alignment failures and .241 of corrected-score failures satisfy none of the prespecified flags. Their within-category failure rates, .067 and .039, are substantially below the respective overall rates but remain nonzero. We therefore interpret the diagnostic decomposition as identifying empirically difficult regimes and not as an exhaustive causal explanation of negative transfer.

REFERENCES

Gulrajani, I. and Lopez-Paz, D. (2021). In search of lost domain generalization. *International Conference on Learning Representations*.

He, Y., Hu, Y., Lin, Y., Zhang, T., and Zhao, H. (2024). Localize-and-Stitch: Efficient model merging via sparse task arithmetic. *Transactions on Machine Learning Research*.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv:1907.11692*.

Lu, Z., Fan, C., Wei, W., Qu, X., Chen, D., and Cheng, Y. (2024). Twin-Merging: Dynamic integration of modular expertise in model merging. *Advances in Neural Information Processing Systems*, 37.

Matena, M. S. and Raffel, C. (2022). Merging models with Fisher-weighted averaging. *Advances in Neural Information Processing Systems*, 35.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21, 1-67.

Tam, D., Bansal, M., and Raffel, C. (2024). Merging by matching models in task parameter subspaces. *Transactions on Machine Learning Research*.

Wang, K., Dimitriadis, N., Ortiz-Jiménez, G., Fleuret, F., and Frossard, P. (2024). Localizing task information for improved model merging and compression. *Proceedings of the 41st International Conference on Machine Learning*, 235, 50268-50287.

Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., and Schmidt, L. (2022). Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. *Proceedings of the 39th International Conference on Machine Learning*, 162, 23965-23998.

Yadav, P., Tam, D., Choshen, L., Raffel, C., and Bansal, M. (2023). TIES-Merging: Resolving interference when merging models. *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 70309-70329.

Yang, E., Shen, L., Wang, Z., Guo, G., Chen, X., Wang, X., and Tao, D. (2024a). Representation surgery for multi-task model merging. *International Conference on Machine Learning*.

Yang, E., Wang, Z., Shen, L., Liu, S., Guo, G., Wang, X., and Tao, D. (2024b). AdaMerging: Adaptive model merging for multi-task learning. *International Conference on Learning Representations*.