TASK SINGULAR VECTORS GO FUNCTIONAL: MODEL MERGING IN THE DATA METRIC

Anonymous authors
Paper under double-blind review

ABSTRACT

Spectral model merging compresses task vectors and reduces interference by orthogonalizing their retained directions. Existing methods make both decisions in the Euclidean geometry of weight space, however, directions that are orthogonal in parameter space can still overlap in the activations the model actually sees. We derive a *joint least-squares* perspective that unifies several existing merging methods in a common objective, and use it to replace Euclidean geometry with a data-induced metric estimated from a few unlabeled samples. Functional TSV (F-TSV) utilizes this metric to choose each task's low-rank subspace and to define cross-task orthogonality, while preserving the expert update within the chosen subspace. Our derivation predicts a square-root metric weighting, and validation sweeps across vision, GLUE, and generative merging select the same setting. Ablations reveal the main empirical effect: data-aware cross-task orthogonalization accounts for almost all of the gains in vision, whereas data-aware subspace selection contributes minimally outside GLUE. F-TSV is training-free, relies only on label-free metric, and produces a single checkpoint with no inference-time overhead, and it consistently improves over its Euclidean counterpart across all reported benchmarks.

1 INTRODUCTION

Finetuning a large pretrained model on different tasks produces a collection of specialist models. Model merging asks whether their task-specific knowledge can be combined into a single network that performs well across all tasks, without retraining or re-examining the labeled data (Wortsman et al., 2022; Ilharco et al., 2023). The central difficulty is interference between tasks: as more experts are merged, updates useful for one task can conflict with the directions needed by another. The central abstraction in model merging is the *task vector* $\tau_t = \theta_t - \theta_0$, the representation of an expert as a displacement of task t's finetuned weights θ_t from the shared pretrained model θ_0. Merging is therefore not simply a matter of adding task vectors, but of deciding which parts of each update should survive and which retained directions should be allowed to coexist. Recent spectral merging methods exploit the observation that these updates are approximately low-rank when viewed layer-by-layer (Hu et al., 2022). TSV-M (Gargiulo et al., 2025) retains a small number of singular directions from each task and then orthogonalizes the retained factors across tasks before recomposition to remove cross-task interference. Isotropic merging (Iso-C, Marczak et al. (2025)) similarly operates on the spectrum of the merged update, flattening it. Despite their different constructions, these methods make the same

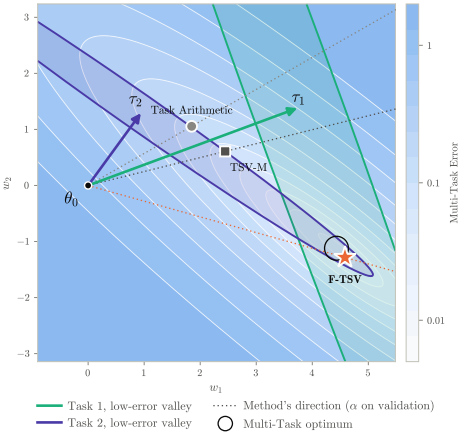

Figure 1: Multi-Task loss landscape depicting final checkpoints after the corresponding model merging method in a 2D setting with 2 tasks. The arrows depict the two task vectors pointing to the low-error valley for their task. The dotted line is the direction each merging method chooses, and on that line, at the α value, is the final point.

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.

two decisions: *which directions of each update to retain*, and *when two retained directions count as interfering*. In both cases, the decisions are based on the Euclidean inner product of weight space.

Euclidean geometry is convenient, but it does not necessarily reflect the function that the network actually computes. For a linear layer, the Euclidean top-k singular directions identify directions along which the weights activate most for an arbitrary unit input. The data, however, may rarely excite those directions. Likewise, Euclidean orthogonality removes correlation in parameter coordinates, but two parameter-space directions that satisfy $v^\top v' = 0$ can still produce strongly correlated changes on the inputs encountered by the model. The relevant overlap is instead determined by the data covariance $C = \mathbb{E}[xx^\top]$. The issue is therefore not that spectral merging lacks a notion of geometry, but that it uses the parameters' geometry rather than the one induced by the data. Figure 1 clearly depicts this issue, showing how cross-task orthogonalization in the data metric helps in pointing towards a common minimum. We make this geometry explicit through a joint least-squares view of model merging. Starting from a second-order approximation of the task losses, we obtain a common objective in which the merged model is penalized for deviating from each expert according to a task-dependent curvature H_t. This perspective places several existing merging methods in the same framework under appropriate choices of curvature and constraints. We use this observation to replace the Euclidean geometry of spectral merging with a *data-induced metric*. **Functional TSV** (**F-TSV**) uses the input metric to make the same two decisions as the other spectral merging methods (Gargiulo et al., 2025; Marczak et al., 2025; Li et al., 2026; An et al., 2025; Cheng et al., 2025), in the geometry of the data: it selects the rank-k subspace with the greatest data-weighted energy for task updates, and defines cross-task orthogonality using the pooled covariance of the task inputs.

We also introduce a whitening power p that controls how the data metric enters the spectral construction. The Euclidean TSV-M geometry (Gargiulo et al., 2025) corresponds to $p = 0$, while PAVE's ranking (An et al., 2025) corresponds to $p = 1$, and the joint least-squares derives the intermediate value $p = \frac{1}{2}$, This prediction is confirmed by held-out sweeps across vision, language, and generative merging, which select the same value. The ablation also clarifies which part of the data-aware decisions matters: in vision, cross-task orthogonalization accounts for almost all of the improvement, while subspace selection contributes only marginally. This also extends to language and generative merging, where the relative contributions differ but the underlying geometric construction remains applicable. The distinction is informative because it separates two mechanisms that are often conflated in spectral merging: identifying useful directions within each task, and preventing the retained directions of different tasks from interfering. F-TSV requires only a small amount of unlabeled data to estimate these metrics. It is training-free, uses no task labels or task losses, and produces a single static checkpoint without inference-time routing or additional computation. The method, therefore, retains the computational structure of spectral merging while replacing its Euclidean notion of similarity with one grounded in the activations the network actually encounters. To summarize, our contributions are fourfold:

- We introduce F-TSV, a data-aware spectral merger (§4.1) that performs subspace selection and cross-task orthogonalization in a data-induced metric.
- We derive a unified joint least-squares formulation that places several existing merging methods in a common objective and motivates the use of the data-induced metric (§4).
- We derive the whitening power $p = \frac{1}{2}$, which interpolates between TSV-M's geometry and the PAVE's ranking; held-out validation confirms this value across all benchmarks (§5).
- We introduce a label-free downstream-sensitivity extension based on $G = \mathbb{E}[J^\top J]$ from Gaussian vector, Jacobian probes (§4.2), we show that the input metric can be estimated using only a small subset of unlabeled samples per task.

2 Related work

Model merging began as weight averaging: model soups average finetuned models that share a pretrained initialization, without adding inference cost (Wortsman et al., 2022), and task arithmetic represents each expert as a task vector τ_t and scales and sums them (Ilharco et al., 2023). Summed task vectors interfere as tasks accumulate, motivating methods that modify the updates at the level of parameters before aggregation. TIES trims small coordinates and resolves sign conflicts (Yadav et al., 2023), DARE drops and rescales them (Yu et al., 2024), and TALL masks localize each task's information in binary masks over the merged vector (Wang et al., 2024). A complementary view is to

exploit the observation that finetuning updates are often closer to low-rank than to coordinate-sparse, particularly layer by layer (Hu et al., 2022; Gargiulo et al., 2025), so spectral methods compress using an SVD rather than a mask. TSV-M keeps a few singular directions per task and orthogonalizes the stacked factors across tasks (Gargiulo et al., 2025). Iso-C flattens the spectrum of the summed update, and Iso-CTS adds task-specific subspaces to the shared one (Marczak et al., 2025).

A parallel line of work weights the merge by each task's curvature. Fisher merging uses the diagonal Fisher (Matena & Raffel, 2022a), RegMean matches each expert's layer outputs on its own inputs using Gram matrices (Jin et al., 2023), and MaTS solves the corresponding richer-curvature system by conjugate gradients (Tam et al., 2024). Daheim et al. (2024) derives averaging, task arithmetic, and Fisher merging from a second-order expansion under different Hessian assumptions, while Tam et al. (2026) compares these methods under a common protocol, specifying data and compute requirements for each. Our joint least-squares formulation (§4) extends to spectral merging, borrowing the Kronecker-factored metric (Martens & Grosse, 2015) while keeping the per-task low-rank structure that exact curvature solvers abandon. Closest to F-TSV are spectral methods influenced by activations. PAVE purifies each expert using its input covariance (An et al., 2025); it truncates the finetuned weight rather than the task vector, un-whitens the kept content by the inverse covariance, and has no cross-task orthogonalization. ESM restricts each update inside the principal subspace of the output changes it induces (Li et al., 2026), under an unconditioned metric, this corresponds to the exact rank-k minimizer of our selection criterion at $p{=}\frac{1}{2}$, but its subsequent orthogonalization remains Euclidean, making data-aware the step our ablation finds least useful (§5). WUDI- and ACE-Merging estimate input statistics directly from the task vectors (Cheng et al., 2025; Xu et al., 2026), while other methods use activations to set merging coefficients (Heyrani Nobari et al., 2026; Yao et al., 2026) or to route among task subspaces at inference time (Crisostomi et al., 2026). F-TSV uses activations only to define the inner product in which *both* spectral decisions are made.

Our geometric interpretation is also related to functional maps in geometry processing. Functional maps represent correspondences between shapes as linear operators between function spaces, expressed in truncated eigenbases that are orthonormal under an area-weighted inner product (Ovsjanikov et al., 2012; 2016). Here, a task update is an operator between feature spaces, weight coordinates are the sampling, and $\langle \Delta, \Delta' \rangle_C = \mathbb{E}_x \langle \Delta x, \Delta' x \rangle$ is the area-weighted inner product in which both truncation and orthogonality are defined (Appendix K). Spectral merging already truncates and orthogonalizes; what it lacks is this functional inner product induced by the data.

3 Background

Given a pretrained model θ_0 and experts $\theta_1, \ldots, \theta_T$ finetuned from it, *merging* them produces a new set of weights $\theta = \theta_0 + \alpha\Delta^\star$, where $\Delta^\star$ aggregates the task vectors $\tau_t = \theta_t - \theta_0$, and α selected on a held-out validation set, as in Task Arithmetic, Tall-Mask, and TSV-M (Ilharco et al., 2023; Wang et al., 2024; Gargiulo et al., 2025). In our setting, we are given an additional small *unlabeled* subset of N inputs from each task's input distribution $\mathcal{D}_t$. No labels are required, and therefore, neither task losses nor task-loss gradients are needed. Since aggregation is usually performed independently at each layer, for a weight matrix $W \in \mathbb{R}^{m\times n}$, we write $\Delta_t \in \mathbb{R}^{m\times n}$ for the corresponding layer block of the task vector τ_t (Table 5). Spectral merging, as introduced by TSV-M (Gargiulo et al., 2025), treats each layer update as approximately low-rank and decomposes $\Delta_t = U_t S_t V_t^\top$. Its construction proceeds in three steps: *Compression* keeps the top $k = \max(1, \lfloor \min(m,n)/T \rfloor)$ singular triplets of the SVD from each task. This budget allocates the layer's spectrum equally among the T experts, so that the retained blocks fill the spectrum exactly once. *Stacking* collects the retained factors from all tasks, $\hat{U} = [U_1^{(k)} \cdots U_T^{(k)}]$, $\hat{V} = [V_1^{(k)} \cdots V_T^{(k)}]$, and stacks the retained singular values into $\hat{s}$. *Interference removal* then replaces the stacked factors by their nearest orthonormal matrices, $\mathrm{polar}(A) := UV^\top$ for $A = USV^\top$, the resulting merged update is

$$\Delta^\star_{\text{TSV-M}} = \mathrm{polar}(\hat{U})\ \mathrm{diag}(\hat{s})\ \mathrm{polar}(\hat{V})^\top. \tag{1}$$

These three steps implicitly make two geometric decisions. Compression determines *which subspace of each task update is retained*, while interference removal determines *when retained directions from different tasks are considered non-interfering*. Here, selecting the "best" rank-k block and orthonormalizing are both defined in the Euclidean metric: two directions are non-interfering when $v^\top v' = 0$. TSV-M, therefore, selects and orthogonalizes directions according to their geometry in *parameter space*, independently of the inputs on which those directions will act.

4 METHOD: MERGING AS JOINT LEAST SQUARES

The polar step of TSV-M drives $(V_t^{(k)})^\top V_{t'}^{(k)}$ to 0, for every pair of tasks $t \neq t'$, removing correlation in weight coordinates. We argue that two retained input directions v, v' that are orthogonal in weight coordinates can still overlap *under the data*: $\mathbb{E}_x[(v^\top x)((v')^\top x)] = v^\top \bar{C} v'$ with $\bar{C}$ the pooled input second moment, and the Euclidean metric has no access to that overlap. To resolve this inconsistency, we developed a framework that starts by expanding each task loss to a second-order approximation around its own expert, and regularizing the merged model towards the pretrained checkpoint, yielding the following *joint least-squares* problem:

$$\min_{\theta} \ \tfrac{1}{2} \sum_{t=1}^{T} (\theta - \theta_t)^\top H_t (\theta - \theta_t) \ + \ \mu\, \Omega(\theta - \theta_0), \tag{2}$$

where $H_t \succeq 0$ is a measure of curvature of the task t, and Ω a regularizer constraining the merged update. Writing $\theta = \theta_0 + \Delta$, the data term is $\sum_t (\Delta - \tau_t)^\top H_t (\Delta - \tau_t)$, and the merged *update* is encouraged to remain close to every expert, each in its own data geometry, by the regularizer. The choices of H_t and Ω, place existing methods under a single framework. When fixing $H_t = I$ and $\Omega = 0$ we obtain Task Arithmetic (Ilharco et al., 2023); $H_t = \mathrm{diag}(F_t)$, the diagonal Fisher, gives Fisher merging (Matena & Raffel, 2022b); $H_t = C_t \otimes I$, with C_t the input second moment, we find RegMean (Jin et al., 2023); with H_t set to the Kronecker-factored Fisher returns MaTS (Tam et al., 2024); and $H_t = I$ with a per-task rank constraint on Δ, solved approximately by the polar step, gives, ultimately, TSV-M (Gargiulo et al., 2025). Table 6 places each method on a ladder of approximations to equation 2.

Choosing the task geometry. Based on our theoretical framework, F-TSV is designed with the suitable geometry that merging requires. H_t should measure how much a weight change alters the network's *function*, not its coordinates. Consider a linear layer $y = Wx$ inside expert t, and let f denote the network's final representation. Expanding the task loss to second order gives, for that layer, the Gauss–Newton curvature

$$H_t = \mathbb{E}_{x \sim \mathcal{D}_t}\big[(xx^\top) \otimes (J^\top J)\big] \ \approx \ C_t \otimes G_t, \qquad J := \partial f / \partial y, \tag{3}$$

so the data enters on one side through x, and the rest of the network enters on the other through the Jacobian J, which carries a perturbation of y to the final representation. H_t is too large to store, so K-FAC (Martens & Grosse, 2015) assumes the input and output sides decouple, leaving one small matrix per side: $C_t = \mathbb{E}_{x \sim \mathcal{D}_t}[xx^\top]$ and $G_t = \mathbb{E}_{x \sim \mathcal{D}_t}[J^\top J]$. The per-layer data term of equation 2 then becomes a *two-sided* norm:

$$(\Delta - \Delta_t)^\top H_t (\Delta - \Delta_t) = \big\| G_t^{1/2} (\Delta - \Delta_t) C_t^{1/2} \big\|_F^2. \tag{4}$$

Each factor has a plain reading: multiplying on the right by $C^{1/2}$ converts "change of weights" into "change of the layer's outputs on task data", and multiplying on the left by $G^{1/2}$ weights how much of that output change survives to the *final representation*:

$$\text{F-TSV} := \underbrace{\mathbb{E}_x \|\Delta x\|^2}_{\text{output change}} = \|\Delta C^{1/2}\|_F^2, \qquad \text{F-TSV}^2 := \underbrace{\mathbb{E}_x \|J \Delta x\|^2}_{\text{representation change}} \ \approx \|G^{1/2} \Delta C^{1/2}\|_F^2. \tag{5}$$

The square-root exponents on C and G are the weighting prescribed by the least-squares geometry. In what follows, we generalize it to a free power p, with $p = \frac{1}{2}$ as the derived default, and study its effect in §5. The Euclidean merging is the case $p = 0$, equivalently $C = G = I$: every parameter direction is treated equally, regardless of how the data excites or propagates it.

Choosing the regularizer. The second ingredient is the constraint Ω. Finetuning updates are sparse, not so much in coordinates (with few non-zero entries), the prior behind TIES and DARE (Yadav et al., 2023; Yu et al., 2024), as in the *spectrum*: they concentrate in few singular directions (Gargiulo et al., 2025). We therefore take Ω to be TSV's rank constraint: the merged update decomposes into one block per task, $\Delta = \sum_{t=1}^{T} \Delta_t^{(k)}$, with $\mathrm{rank}(\Delta_t^{(k)}) \leq k$, so that the T blocks together fill the layer's spectrum exactly once. Each $\Delta_t^{(k)}$ holds what survives of expert t's update Δ_t; *which* part survives is what F-TSV method decides (§4.1).

Why not solve equation 2 exactly. With $\Omega = 0$ and the one-sided metric, the exact minimizer of equation 2 is RegMean's $\Delta = (\sum_t \Delta_t C_t)(\sum_t C_t)^{-1}$ (Jin et al., 2023), which multiplies expert

content by the full inverse of a *sum of estimated* covariances, amplifying precisely the directions the estimates resolve worst. It also discards the per-task low-rank structure that denoises each expert update. We keep the constrained spectral form and use the data metric only where it is safe. The design principle is simple: the metric *chooses* the subspaces and *defines* orthogonality, restricting the expert update rather than reweighting it. Therefore, F-TSV should not be interpreted as the exact minimizer of equation 2. Rather, the joint least-squares objective motivates the metric, while the spectral and orthogonality constraints turn it into a tractable merger, as in Algorithm 1.

4.1 FUNCTIONAL TSV

Step 1: metric-guided selection. The Euclidean choice of $\Delta_t^{(k)}$ keeps the directions v for which $\|\Delta_t v\|$ is largest over unit vectors. The functional choice maximizes the same quantity, in the data geometry, over vectors of unit C_t^{2p}-norm, so the directions the data actually excites are easier to keep. We (i) take the SVD of the whitened update $\Delta_t C_t^p = \tilde{U}\tilde{S}\tilde{V}^\top$ and retains its top-k right singular vectors $\tilde{V}^{(k)}$; (ii) we map them back to weight coordinates and orthonormalize them: $Q_t = \mathrm{QR}(C_t^p \tilde{V}^{(k)})$; and (iii) keep the expert's action on that subspace *undistorted*:

$$\Delta_t^{(k)} = \Delta_t \, Q_t Q_t^\top, \tag{6}$$

We store the block in SVD form via $\Delta_t Q_t = U_t^{(k)} S_t^{(k)} (V_t^{(k)})^\top$, these are the factors that will be stacked and orthogonalized in Step 2, so the block's singular values are genuine singular values of Δ_t restricted to the subspace, the exact rank-k minimizer of the one-sided functional approximation error equation 5 is the orthogonal projection $\tilde{U}^{(k)}(\tilde{U}^{(k)})^\top \Delta_t$ onto the leading left singular directions of the whitened update. The equation 6 is equal to this exact block followed by a further projection onto $\mathrm{range}(Q_t)$: $\Delta_t Q_t Q_t^\top = (\tilde{U}^{(k)}(\tilde{U}^{(k)})^\top \Delta_t) Q_t Q_t^\top$. Consequently, equation 6 is weakly suboptimal under the isolated functional approximation criterion, this solution is optimal only when the task is considered in isolation. By retaining the selected input directions themselves as the block's right factors, the next step can measure and remove *cross-task* interference precisely within the subspace selected by the data-aware spectrum. The exact minimizer shares the same column space but generally has a different row space. The selection criterion alone, therefore, does not determine which block is preferable once the subsequent cross-task orthogonalization is taken into account.

Step 2: interference removal. Two selected directions v, v' from different tasks can still respond to the same inputs, and their overlap is determined by the pooled input metric:

$$\mathbb{E}_x\big[(v^\top x)((v')^\top x)\big] = v^\top \bar{C} v', \qquad \bar{C} = \tfrac{1}{T}\textstyle\sum_t \mathrm{unit}(C_t), \tag{7}$$

with $\mathrm{unit}(\cdot)$ the rescaling to unit mean eigenvalue prevents tasks with larger activation from dominating the shared geometry. The pooled metric $\bar{C}$ is then conditioned using equation 10. We redefine the notion of non-interference as $\bar{C}$-orthonormality and replace the stacked right factor by the nearest $\bar{C}^{\,2p}$-orthonormal matrix in the norm $\|(\cdot)\,\bar{C}^{\,p}\|_F$, the metric analogue of the polar step[1]

$$R = \mathrm{polar}\big(\hat{V}^\top \bar{C}^{\,p}\big)\,\bar{C}^{\,-p}, \qquad R\,\bar{C}^{\,2p} R^\top = I, \tag{8}$$

which at the derived $p = \frac{1}{2}$ is exactly the $\bar{C}$-orthonormality of equation 7. The left factor keeps $L = \mathrm{polar}(\hat{U})$ and the merged update is $\Delta^\star = L\,\mathrm{diag}(\hat{s})R$, for an in-depth view see Algorithm 1. So, F-TSV changes only the *geometry* in which spectral merging operates: the data metric determines which directions are retained and which retained directions must be orthogonalized to reduce their interference, while the singular magnitudes $\hat{s}$ of the retained expert content are preserved.

4.2 TWO-SIDED FUNCTIONAL TSV EXTENSION

The input metric asks which changes of a layer's output are large on task data, but it does not distinguish between changes that are subsequently preserved and those that are attenuated by the rest of the network. F-TSV^2 closes this gap by introducing the output-side sensitivity $G_t = \mathbb{E}[J^\top J]$.

For a Gaussian probe $v \sim \mathcal{N}(0, I)$ drawn in representation space, $\mathbb{E}_v[(J^\top v)(J^\top v)^\top] = J^\top J$, a Hutchinson-style identity (Hutchinson, 1989), so the second moment of vector-Jacobian products

[1] Writing $R' = R\,\bar{C}^{\,p}$ turns the constraint into $R'R'^\top = I$ and the objective into $\|R' - \hat{V}^\top \bar{C}^{\,p}\|_F$, whose minimizer is $\mathrm{polar}(\hat{V}^\top \bar{C}^{\,p})$.

Table 1: **Vision benchmark results.** Average absolute accuracy, subscript in parentheses is the normalized average accuracy. **Bold** is the best value per column over all rows.

	Method	ViT-B-32			ViT-B-16			ViT-L-14		
		8 tasks	14 tasks	20 tasks	8 tasks	14 tasks	20 tasks	8 tasks	14 tasks	20 tasks
Base	Zeroshot	$48.1_{(54.8)}$	$57.0_{(64.5)}$	$57.5_{(65.2)}$	$55.4_{(60.7)}$	$62.0_{(68.0)}$	$62.5_{(68.3)}$	$64.9_{(69.2)}$	$69.1_{(73.8)}$	$68.3_{(72.7)}$
	Finetuned	$90.3_{(100.0)}$	$89.1_{(100.0)}$	$89.5_{(100.0)}$	$92.4_{(100.0)}$	$91.4_{(100.0)}$	$91.9_{(100.0)}$	$94.2_{(100.0)}$	$93.4_{(100.0)}$	$94.0_{(100.0)}$
Fixed	Weight Averaging	$67.1_{(74.6)}$	$65.5_{(73.8)}$	$64.4_{(72.6)}$	$73.0_{(78.8)}$	$70.8_{(77.3)}$	$69.2_{(75.3)}$	$80.5_{(85.3)}$	$78.5_{(83.8)}$	$76.1_{(80.9)}$
	Task Arithmetic	$70.4_{(77.9)}$	$66.8_{(75.2)}$	$64.0_{(71.9)}$	$75.3_{(81.1)}$	$72.0_{(78.6)}$	$69.0_{(75.0)}$	$84.4_{(89.3)}$	$80.4_{(85.8)}$	$76.8_{(81.7)}$
	TSV-M	$83.4_{(92.0)}$	$79.1_{(88.6)}$	$76.8_{(85.9)}$	$85.8_{(92.5)}$	$81.9_{(89.3)}$	$80.0_{(86.9)}$	$91.2_{(96.7)}$	$88.8_{(94.9)}$	$87.5_{(93.0)}$
	Iso-C	$83.7_{(92.5)}$	$78.7_{(88.3)}$	$74.9_{(83.9)}$	$87.6_{(94.6)}$	$81.7_{(89.1)}$	$78.0_{(84.8)}$	$92.4_{(97.9)}$	$89.9_{(96.1)}$	$86.8_{(92.3)}$
	Iso-CTS	$84.3_{(93.2)}$	$80.7_{(90.5)}$	$76.9_{(86.1)}$	$88.7_{(95.8)}$	$84.5_{(92.2)}$	$81.0_{(88.0)}$	$93.1_{(98.8)}$	$90.5_{(96.8)}$	$89.4_{(95.1)}$
Data	ESM	$84.4_{(93.3)}$	$80.1_{(89.9)}$	$78.9_{(88.3)}$	$87.6_{(94.6)}$	$82.9_{(90.6)}$	$81.3_{(88.3)}$	$92.0_{(97.6)}$	$89.2_{(95.4)}$	$88.9_{(94.5)}$
	RegMean	$82.5_{(91.0)}$	$77.2_{(86.5)}$	$75.4_{(84.3)}$	$85.3_{(92.0)}$	$80.6_{(87.8)}$	$78.0_{(84.7)}$	$91.0_{(96.5)}$	$88.1_{(94.1)}$	$85.6_{(90.9)}$
	Func. TSV	$86.8_{(96.0)}$	$83.6_{(93.7)}$	$82.2_{(91.8)}$	$\mathbf{89.1_{(96.4)}}$	$85.6_{(93.6)}$	$85.0_{(92.3)}$	$93.2_{(98.9)}$	$91.5_{(97.9)}$	$\mathbf{91.9_{(97.7)}}$
	Func. TSV^2	$\mathbf{87.0_{(96.2)}}$	$\mathbf{83.8_{(93.9)}}$	$\mathbf{82.4_{(92.0)}}$	$88.7_{(96.0)}$	$\mathbf{86.1_{(94.1)}}$	$\mathbf{85.6_{(93.0)}}$	$\mathbf{93.3_{(99.0)}}$	$\mathbf{91.6_{(98.0)}}$	$91.8_{(97.6)}$

Table 2: **GLUE benchmark.** Headline score per task (Matthews for CoLA, Spearman for STS-B, accuracy elsewhere), subscript in parentheses indicates the normalized score. **Bold** is the best method per column, with α representing the validation-selected coefficient.

	Method	RoBERTa-base — GLUE								Avg.	α
		CoLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE		
Base	Pre-trained	$2.6_{(4.7)}$	$50.9_{(53.8)}$	$74.8_{(85.0)}$	$0.0_{(0.0)}$	$31.6_{(35.3)}$	$32.4_{(37.0)}$	$49.6_{(53.7)}$	$47.3_{(71.2)}$	$36.1_{(42.6)}$	–
	Finetuned	55.1	94.7	88.0	87.7	89.5	87.5	92.4	66.4	82.7	–
Fixed	Model Soup	$2.6_{(4.7)}$	$55.3_{(58.4)}$	$75.5_{(85.8)}$	$42.0_{(47.9)}$	$40.0_{(44.7)}$	$41.8_{(47.7)}$	$59.8_{(64.7)}$	$47.3_{(71.2)}$	$45.5_{(53.1)}$	–
	Task Arithmetic	$2.6_{(4.7)}$	$85.6_{(90.3)}$	$78.9_{(89.7)}$	$26.2_{(29.8)}$	$72.9_{(81.5)}$	$62.3_{(71.2)}$	$76.2_{(82.5)}$	$52.3_{(78.8)}$	$57.1_{(66.1)}$	0.4
	TSV-M	$21.5_{(39.0)}$	$92.8_{(97.9)}$	$78.1_{(88.8)}$	$43.9_{(50.1)}$	$82.8_{(92.5)}$	$76.1_{(87.0)}$	$83.6_{(90.4)}$	$52.0_{(78.3)}$	$66.4_{(78.0)}$	1.4
	Iso-C	$26.8_{(48.7)}$	$89.6_{(94.6)}$	$78.2_{(88.9)}$	$58.2_{(66.4)}$	$84.4_{(94.4)}$	$76.2_{(87.1)}$	$83.5_{(90.3)}$	$\mathbf{61.7_{(92.9)}}$	$69.8_{(82.9)}$	3.1
	Iso-CTS	$40.5_{(73.5)}$	$84.4_{(89.1)}$	$78.8_{(89.6)}$	$49.0_{(55.8)}$	$82.6_{(92.3)}$	$67.7_{(77.4)}$	$78.8_{(85.2)}$	$57.4_{(86.4)}$	$67.4_{(81.2)}$	3.4
Data	ESM	$34.1_{(61.8)}$	$70.8_{(74.7)}$	$75.6_{(85.9)}$	$34.9_{(39.8)}$	$63.6_{(71.1)}$	$63.1_{(72.2)}$	$51.7_{(55.9)}$	$47.3_{(71.2)}$	$55.1_{(66.6)}$	0.5
	RegMean	$44.7_{(81.1)}$	$\mathbf{93.9_{(99.2)}}$	$80.7_{(91.7)}$	$62.1_{(70.8)}$	$82.9_{(92.7)}$	$79.7_{(91.1)}$	$\mathbf{87.9_{(95.1)}}$	$41.2_{(62.0)}$	$71.6_{(85.5)}$	1.4
	Func. TSV	$45.4_{(82.4)}$	$92.5_{(97.7)}$	$\mathbf{82.1_{(93.4)}}$	$58.6_{(66.8)}$	$\mathbf{84.5_{(94.4)}}$	$\mathbf{81.3_{(92.9)}}$	$85.7_{(92.7)}$	$54.9_{(82.6)}$	$73.1_{(87.9)}$	3.0
	Func. TSV^2	$\mathbf{46.1_{(83.7)}}$	$92.3_{(97.5)}$	$80.0_{(90.9)}$	$\mathbf{63.5_{(72.4)}}$	$84.1_{(94.0)}$	$80.0_{(91.4)}$	$86.3_{(93.4)}$	$58.8_{(88.6)}$	$\mathbf{73.9_{(89.0)}}$	3.2

estimates G_t without materializing J. In practice, a single forward pass per batch stores the output y of each linear layer (the *sites*). Each of n_p probes then costs a single backward pass, which produces $g = J^\top v$ at all sites at once. Accumulating $gg^\top$ estimate G_t without labels or a task loss, making the estimator entirely label-free. Selection now whitens on both sides, $G_t^p \Delta_t C_t^p = \tilde{U}\tilde{S}\tilde{V}^\top$, with Q_t and $\Delta_t^{(k)}$ constructed as in step 1. This changes the interpretation of the spectrum as: a direction scores highly only when the data excites it *and* the layers above propagate its effect, no separate output-side truncation is needed (Lemma 1). The same geometry also defines interference on the output side, mirroring equation 7 with $\langle u, u' \rangle_{\bar{G}} = \mathbb{E}_x \langle J(x)u, J(x)u' \rangle$ and $\bar{G} = \frac{1}{T}\sum_t \mathrm{unit}(G_t)$, we orthonormalize the stacked left factors in the pooled sensitivity metric,

$$L = \bar{G}^{-p}\,\mathrm{polar}\big(\bar{G}^p \hat{U}\big), \qquad L^\top \bar{G}^{2p} L = I, \tag{9}$$

which is the exact left-sided counterpart of equation 8. Algorithm 1 completes the two-sided construction, setting $G_t = I$ removes the downstream sensitivity and recovers the one-sided F-TSV exactly. The one-sided method is the main practical technique, F-TSV^2 explores whether downstream sensitivity further improves the same geometric framework.

5 Experiments & Results

The experiments test the geometric claim underlying F-TSV, each experiment below answers a specific question. We ask whether moving spectral merging from Euclidean weight space into the data metric improves the merge (**Q1**), which of the two data-aware decisions, subspace selection or cross-task orthogonalization, accounts for the improvement (**Q2**), where in the network the relevant geometry resides (**Q3**), whether the theoretically derived whitening power is recovered empirically (**Q4**), and whether the same geometry extends to generative merging (**Q5**).

Benchmarks setup. We evaluate the same geometric construction across three settings. For *vision:* we used CLIP encoders (ViT- B/32, B/16, L/14) merged over the 8-task suite (Ilharco et al., 2023) and its 14- and 20-task extensions (Wang et al., 2024), evaluated with frozen per-task heads. For *language* we opted for RoBERTa-base experts on 8 GLUE tasks (Wang et al., 2018), following the benchmark's task-specific metrics: Matthews for CoLA, Spearman for STS-B, and accuracy elsewhere. For *generative* we used five Llama-3.2-3B MergeBench experts (He et al., 2026) and report the benchmark's five-pillar score.

Protocol. For every method with a merge coefficient, we select α on a held-out validation split. On vision and GLUE, the validation split is a seeded 10% subset of each task's training data. On MergeBench, where evaluating each candidate α is substantially more expensive, we select α based on the teacher-forced validation likelihood. The reported results are from the test split evaluated once at the selected validation α. We report the absolute and the *normalized* score, *i.e.* the merged score divided by the corresponding expert and averaged across tasks.

Main results (Q1). We first ask whether replacing Euclidean spectral geometry with the data metric improves the merged model. On *vision* (Table 1), the F-TSV family obtains the highest score among the evaluated methods in all nine settings, improving normalized accuracy over the best merging methods. The margin over TSV-M, the $p = 0$ Euclidean case, increases with task count across all backbones. This scaling is consistent with the role of cross-task interference: as more tasks compete for the same spectral budget, the value of defining the cross-task decorrelation under the data geometry increases. Figure 2 shows that this growing margin is associated with cross-task decorrelation rather than with changing which subspace each task retains. On *Language* (Table 2), F-TSV raises the average score to 73.1 (87.9 normalized), while F-TSV^2 reaches 73.9 (89.0 normalized). The strongest baseline, RegMean, surprisingly reaches 71.6 (85.5 normalized), the two curvature-aware families trade places by task: RegMean is ahead on STS-B and QNLI, while F-TSV on MNLI and MRPC. Notably, unlike the vision results, RegMean, a data-driven method, is a weaker baseline on vision (Table 1) than on GLUE; by contrast, our method is more balanced between the two benchmarks.

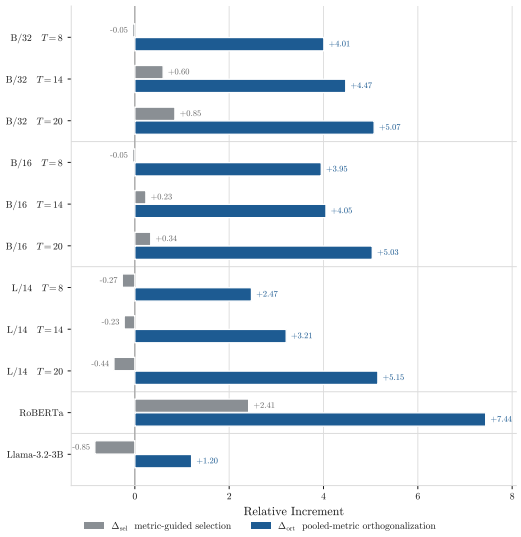

Figure 2: Comparison between metric-guided selection and the pooled-metric orthogonalization. Δ_{sel} and Δ_{ort} are the two increments in normalized points w.r.t. Euclidean polar (TSV).

Ablation (Q2). The central question here is whether the improvement comes from choosing better subspaces for each task, from preventing interference between the retained subspaces, or from both. Figure 2 separates the two data-aware decisions of §4.1 on every benchmark. On vision, the contrast is sharp: averaged over the nine cells, metric-guided selection is worth $+0.11$ normalized points and the pooled-metric orthogonalization $+4.16$. Selection is small and directionless, spanning -0.44 to $+0.85$ and with five *negative* outcomes, whereas orthogonalization improves every setting, and increases with task count. What scaling the number of tasks makes more valuable is the *decorrelation* of the retained subspaces under the data metric, rather than the per-task choice of which subspace to retain. The effect persists across modalities, although the margin does not. On MergeBench, orthogonalization remains the positive contribution ($+1.20$ on average), while selection does not (-0.85). GLUE is the setting where selection contributes substantially: $+2.41$ of the 9.85 normalized improvement points. Across the three modalities, the orthogonalization is consistently positive, whereas selection is positive only on GLUE. This bears directly on per-task purification methods such as PAVE (An et al., 2025), whose ranking rule is close to our metric-guided selection and which have no cross-task stage. In our experiments, cross-task orthogonalization accounts for substantially more of the observed improvement than metric-guided selection. Switching a metric off reproduces the parent method in every reported digit as Proposition 3 prescribes (Appendix I).

Table 3: **Site attribution on vision:** the MLP carries the input geometry, and the output metric is near-flat. Which weight matrices keep their measured metric, each row is that metric's marginal value at those sites, and each none row reproduces its parent. Absolute score with normalized in subscript. *Gain* is the normalized-score change against the parent.

Sites kept	**Input metric** C ViT-B/32, 8 tasks		**Sites kept**	**Output metric** G ViT-B/32, 8 tasks	
	Score	Gain		Score	Gain
none ($\equiv$ TSV-M)	$83.41_{(92.05)}$	—	none ($\equiv$ F-TSV)	$86.81_{(96.02)}$	—
q/k/v	$83.62_{(92.27)}$	$+0.21$	q/k/v	$86.51_{(95.65)}$	-0.37
all attention	$83.90_{(92.61)}$	$+0.55$	all attention	$86.66_{(95.84)}$	-0.18
non-residual	$84.72_{(93.56)}$	$+1.51$	non-residual	$86.82_{(96.02)}$	$+0.00$
residual	$85.78_{(94.74)}$	$+2.68$	residual	$86.68_{(95.86)}$	-0.15
q/k/v + residual	$85.85_{(94.79)}$	$+2.73$	q/k/v + residual	$86.67_{(95.85)}$	-0.17
MLP linears	$86.25_{(95.29)}$	$+3.24$	MLP linears	$\mathbf{86.99_{(96.25)}}$	$+0.23$
MLP + residual	$86.54_{(95.65)}$	$+3.59$	MLP + residual	$86.83_{(96.03)}$	$+0.02$
all sites	$\mathbf{86.81_{(96.02)}}$	$+3.96$	all sites	$86.96_{(96.22)}$	$+0.20$

Where the geometry lives (Q3). We ask *where in the network* the geometry resides, by limiting the metric to a subset of sites. Table 3 and Table 14 show *which* matrices preserve their measured metric, for vision and GLUE. On vision, the input geometry is concentrated in the MLP and in the sites contributing to the residual stream. Keeping C only at the attention projections recovers 0.55 of the available 3.96 points, while the MLP alone recovers 3.24, and MLP plus residual 3.59, with no combination reaching the full gain. The output metric is comparatively flat: peaking at $+0.23$ with the MLP, while several subsets perform slightly *below* the one-sided F-TSV method. On GLUE, the input metric tells a different story: the stronger proper subset is attention q/k/v plus residual, recovering $+7.13$ of the available 9.85. Here, unlike in vision, the MLP alone is not the dominant site, contributing $+4.57$. Adding G produces a minor change: at all sites, it recovers $+1.12$ over the one-sided construction, most of it from the MLP. This ablation suggests that the post-normalization residual streams are anisotropic and task-discriminative across modalities, with modality-dependent contributions from MLP or attention. In practice, collecting statistics at MLP sites alone recovers 82% of the vision gain at lower cost, while on GLUE the best subset recovers 72%.

The whitening power(Q4). The derivation of §4 predicts how strongly the data metric should enter the spectral construction. The whitening power p in $C^{\,p}$ should be $\frac{1}{2}$, the square-root geometry induced by the least-squares objective of equation 5, rather than the $p{=}1$ ranking of PAVE (An et al., 2025) or the $p{=}0$ of TSV (Gargiulo et al., 2025). This is the unique power for which the metric-polar constraint also coincides exactly with the proposed functional orthogonality criterion. Sweeping p on all three modalities gives a grid showing the derived value $p{=}\frac{1}{2}$ holds as the best point:

p	0 (TSV-M)	0.25	0.5 (derived FTSV)	1.0 (PAVE)	2.0
ViT-B/32 $T{=}8$	83.41 (92.05)	86.47 (95.52)	**86.81 (96.02)**	85.92 (95.07)	84.93 (93.94)
RoBERTa	66.35 (78.01)	71.28 (85.29)	**73.13 (87.86)**	68.40 (82.52)	56.60 (68.16)
Llama-3.2-3B	39.37	39.67	**39.76**	36.46	13.85

The improvement of $p{=}\frac{1}{2}$ over $p{=}1$ is 0.95 normalized points on vision, 5.34 on GLUE, and 3.30 on MergeBench. The optimum is interior in every setting not at either endpoint. On the generative benchmark, however, the margin over $p{=}\frac{1}{4}$ is only 0.09 points, indicating an agreement with the theoretical prediction rather than as a strong separation from neighboring powers. Higher powers do not simply strengthen data geometry: the pooled orthogonalization applies $\mathrm{polar}(M\,\bar{C}^{\,p})\,\bar{C}^{-p}$ and hence they increasingly relies on inverse powers of the conditioned metric as p grows, with amplification $\kappa(\bar{C})^{\,p}$ amplifying the effects of estimation and conditioning. Experimentally, selecting p

Table 4: Scores per-pillar, every green row at the derived exponent $p{=}\frac{1}{2}$ and every row at its own validation-selected α. The input metric's gain is concentrated in math, and the output metric's in safety, at the cost of coding for both F-TSV. **Bold** is the best merge per column.

	Method	Llama-3.2-3B — MergeBench					
		Instruct.	Math	Coding	Multiling.	Safety	**Avg.**
Base	Pre-trained	12.68	29.95	27.03	40.13	29.18	27.79
	Finetuned	39.41	57.47	44.23	40.92	76.30	51.66
Fixed	Model Soup	14.75	42.38	37.95	41.81	36.60	34.70
	Task Arithmetic	13.30	41.24	37.89	41.76	37.78	34.39
	TSV-M	20.20	50.42	**40.41**	**41.87**	44.34	39.45
Data	ESM	18.30	53.30	39.56	41.19	44.92	39.45
	Func. TSV	**21.14**	**57.01**	36.46	40.30	44.10	**39.80**
	Func. TSV^2	20.53	56.33	34.53	40.41	**46.84**	39.73

on a held-out validation set rather than by derivation, $\frac{1}{2}$ emerges as optimal on all three modalities. On MergeBench, the validation likelihood is minimized at $p{=}\frac{1}{2}$ (1.640, against 1.641 at $p{=}\frac{1}{4}$ and 1.666 at $p{=}1$). Thus, the value prescribed by the least-squares geometry is also the value selected empirically, providing a direct agreement between derivation and validation.

Generative merging (Q5). Here, we ask whether the same previous constructions carry over to generative merging, where the inputs and downstream computations differ substantially. Table 4 reports five **Llama-3.2-3B** MergeBench experts, scored on all five pillars. At the derived $p = \frac{1}{2}$, F-TSV reaches 39.80 and F-TSV^2 39.73, compared with 39.45 for both TSV-M, ESM. The margin, $+0.35$, is an order of magnitude smaller than on vision or GLUE, so we read it as evidence that the data-induced geometry remains applicable to generative merging rather than as a competitive result. The per-pillar results explain where this transfer comes from: relative to TSV-M, the input metric gains $+6.60$ on math and $+0.94$ on instruction, while giving back 3.95 on coding, 1.57 on multilingual, and 0.24 on safety. Adding the output metric shifts the balance rather than improving it: F-TSV^2 recovers 0.68 of the math gain and 0.61 of the instruction gain, adds $+2.74$ in safety, and incurs a further 1.93 loss in coding. The common pattern is therefore not a uniform improvement across all capabilities, but a redistribution of performance along the geometry captured by the metric. The gains concentrate on the likelihood- and reasoning-oriented pillars, while coding, evaluated through sampled program synthesis, moves in the opposite direction.

6 Conclusion

We recast spectral model merging around two geometric decisions: *which directions of an update to keep*, and *which count as interfering*, as questions about data rather than parameters, and answered both from the change a merge induces on real activations. Functional TSV alters only *which* subspace survives and *what orthogonality means*, preserving the spectral experts' content, and reduces exactly to Euclidean TSV-M when the metric is mathematically switched off.

The ablation separates the benefit between the two data-aware decisions prescribed by the objective. In vision tasks, when averaged across nine settings, three backbones, and 8 to 20 tasks, metric-guided selection gains only $+0.11$ normalized points, compared with $+4.16$ from pooled-metric orthogonalization. Notably, the latter benefits increase with the number of tasks, while the former does not. The main effect of replacing Euclidean geometry is therefore not to identify better directions for each task in isolation, but to prevent the directions retained by different tasks from interfering with the data they actually encounter. This extends beyond vision, the same control in GLUE and in generative merging places orthogonalization ahead in both cases, indicating that this ordering is consistent across different scenarios. The geometry carries to generative merging as well, where both variants place above data-free and data-driven baselines, while the smaller margin suggests that its effect should be understood primarily as a geometric correction rather than a decisive gain.

REFERENCES

Bang An, Yibo Yang, Philip Torr, and Bernard Ghanem. Purifying task vectors in knowledge-aware subspace for model merging, 2025. URL https://arxiv.org/abs/2510.14697.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – Mining Discriminative Components with Random Forests. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 446–461, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10599-4. doi: 10.1007/978-3-319-10599-4_29.

Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote Sensing Image Scene Classification: Benchmark and State of the Art. *Proceedings of the IEEE*, 105(10):1865–1883, October 2017. ISSN 1558-2256. doi: 10.1109/JPROC.2017.2675998. URL https://ieeexplore.ieee.org/document/7891544/?arnumber=7891544. Conference Name: Proceedings of the IEEE.

Runxi Cheng, Feng Xiong, Yongxian Wei, Wanyun Zhu, and Chun Yuan. Whoever started the interference should end it: Guiding data-free model merging via task vectors. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=xR9msNaREW.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing Textures in the Wild. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3606–3613, Columbus, OH, USA, June 2014. IEEE. ISBN 978-1-4799-5118-5. doi: 10.1109/CVPR.2014.461. URL https://ieeexplore.ieee.org/document/6909856.

Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep Learning for Classical Japanese Literature, November 2018. URL http://arxiv.org/abs/1812.01718. arXiv:1812.01718 [cs, stat].

Adam Coates, Andrew Ng, and Honglak Lee. An Analysis of Single-Layer Networks in Unsupervised Feature Learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 215–223. JMLR Workshop and Conference Proceedings, June 2011. URL https://proceedings.mlr.press/v15/coates11a.html. ISSN: 1938-7228.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: Extending MNIST to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 2921–2926, May 2017. doi: 10.1109/IJCNN.2017.7966217. URL https://ieeexplore.ieee.org/document/7966217/?arnumber=7966217. ISSN: 2161-4407.

Donato Crisostomi, Alessandro Zirilli, Antonio Andrea Gargiulo, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, Iacopo Masi, and Emanuele Rodolà. Mass: Moerging through adaptive subspace selection. In *International Conference on Learning Representations*, volume 2026, pp. 19982–20008, 2026.

Nico Daheim, Thomas Möllenhoff, Edoardo M Ponti, Iryna Gurevych, and Mohammad Emtiyaz Khan. Model merging by uncertainty-based gradient matching. In *International Conference on Learning Representations*, volume 2024, pp. 11836–11854, 2024.

Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodola. Task singular vectors: Reducing task interference in model merging. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18695–18705. IEEE, 2025.

Ian J. Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, Yingbo Zhou, Chetan Ramaiah, Fangxiang Feng, Ruifan Li, Xiaojie Wang, Dimitris Athanasakis, John Shawe-Taylor, Maxim Milakov, John Park, Radu Ionescu, Marius Popescu, Cristian Grozea, James Bergstra, Jingjing Xie, Lukasz Romaszko, Bing Xu, Zhang Chuang, and Yoshua Bengio. Challenges in Representation Learning: A Report on Three Machine Learning Contests. In Minho Lee, Akira Hirose, Zeng-Guang Hou, and Rhee Man Kil (eds.), *Neural Information Processing*, pp. 117–124, Berlin, Heidelberg, 2013. Springer. ISBN 978-3-642-42051-1. doi: 10.1007/978-3-642-42051-1_16.

Yifei He, Siqi Zeng, Yuzheng Hu, Rui Yang, Tong Zhang, and Han Zhao. Mergebench: A benchmark for merging domain-specialized llms. *Advances in Neural Information Processing Systems*, 38, 2026.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, July 2019. ISSN 2151-1535. doi: 10.1109/JSTARS.2019.2918242. URL https://ieeexplore.ieee.org/document/8736785/?arnumber=8736785. Conference Name: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

Amin Heyrani Nobari, Kaveh Alimohammadi, Ali ArjomandBigdeli, Akash Srivastava, Faez Ahmed, and Navid Azizan. Activation-informed merging of large language models. *Advances in neural information processing systems*, 38:72975–72998, 2026.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Michael F Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 18(3):1059–1076, 1989.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=FCnohuR6AnM.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pp. 554–561, 2013.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Yann LeCun, Corinna Cortes, Chris Burges, et al. MNIST handwritten digit database, 1998. URL https://yann.lecun.com/exdb/mnist/.

Longhua Li, Lei Qi, Qi Tian, and Xin Geng. Model merging in the essential subspace. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31145–31154, June 2026.

Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D. Bagdanov, and Joost Van De Weijer. No task left behind: Isotropic model merging with common and task-specific subspaces. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 43177–43199. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/marczak25a.html.

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

Michael S Matena and Colin Raffel. Merging models with fisher-weighted averaging. *Advances in Neural Information Processing Systems*, 35:17703–17716, 2022a.

Michael S Matena and Colin Raffel. Merging models with fisher-weighted averaging. *Advances in Neural Information Processing Systems*, 35:17703–17716, 2022b.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. URL http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

Maria-Elena Nilsback and Andrew Zisserman. Automated Flower Classification over a Large Number of Classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pp. 722–729, December 2008. doi: 10.1109/ICVGIP.2008.47. URL https://ieeexplore.ieee.org/document/4756141.

Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (ToG)*, 31(4):1–11, 2012.

Maks Ovsjanikov, Etienne Corman, Michael Bronstein, Emanuele Rodolà, Mirela Ben-Chen, Leonidas Guibas, Frederic Chazal, and Alex Bronstein. Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses*, SA '16, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450345385. doi: 10.1145/2988458.2988494. URL https://doi.org/10.1145/2988458.2988494.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3498–3505, June 2012. doi: 10.1109/CVPR.2012.6248092. URL https://ieeexplore.ieee.org/document/6248092. ISSN: 1063-6919.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In David Yarowsky, Timothy Baldwin, Anna Korhonen, Karen Livescu, and Steven Bethard (eds.), *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL https://aclanthology.org/D13-1170.

Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In *The 2011 International Joint Conference on Neural Networks*, pp. 1453–1460, July 2011. doi: 10.1109/IJCNN.2011.6033395. URL https://ieeexplore.ieee.org/document/6033395/?arnumber=6033395. ISSN: 2161-4407.

Derek Tam, Mohit Bansal, and Colin Raffel. Merging by matching models in task parameter subspaces. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=qNGo6ghWFB.

Derek Tam, Yash Kant, Brian Lester, Igor Gilitschenski, and Colin Raffel. Realistic evaluation of model merging for compositional generalization. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=j7ye0nXvEm.

Bastiaan S. Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation Equivariant CNNs for Digital Pathology. In Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger (eds.), *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pp. 210–218, Cham, 2018. Springer International Publishing. ISBN 978-3-030-00934-2. doi: 10.1007/978-3-030-00934-2_24.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, pp. 353–355, 2018.

Ke Wang, Nikolaos Dimitriadis, Guillermo Ortiz-Jiménez, François Fleuret, and Pascal Frossard. Localizing task information for improved model merging and compression. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pp. 23965–23998. Pmlr, 2022.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms, September 2017. URL http://arxiv.org/abs/1708.07747. arXiv:1708.07747 [cs, stat].

Jianxiong Xiao, Krista A. Ehinger, James Hays, Antonio Torralba, and Aude Oliva. SUN Database: Exploring a Large Collection of Scene Categories. *International Journal of Computer Vision*, 119(1):3–22, August 2016. ISSN 1573-1405. doi: 10.1007/s11263-014-0748-y. URL https://doi.org/10.1007/s11263-014-0748-y.

Bo Xu, Haotian Wu, Hehai Lin, Weiquan Huang, Beier Zhu, Yao Shu, and Chengwei Qin. Ace-merging: Data-free model merging with adaptive covariance estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29472–29482, June 2026.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. *Advances in neural information processing systems*, 36:7093–7115, 2023.

Yuxuan Yao, Shuqi Liu, Zehua Liu, Qintong Li, Mingyang Liu, Xiongwei Han, Zhijiang Guo, Han Wu, and Linqi Song. Activation-guided consensus merging for large language models. *Advances in Neural Information Processing Systems*, 38:124041–124069, 2026.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: absorbing abilities from homologous models as a free lunch. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

REPRODUCIBILITY STATEMENT

All merging methods, the three evaluation pipelines, configurations and sweep definitions are released. §5 states the protocol; Appendix E states the exact conditioning recipe; Appendix Q states the backbones and expert checkpoints, splits, statistics budgets, rank budgets, coefficient grids, observability rules and evaluation harnesses, together with the selected coefficient of every reported cell (Table 16). Statistics are cached keyed by task, model, split and budget, and every ablation cell reuses the same caches, so differences between cells reflect the merging rule alone. Every results file records the resolved merger configuration, the selected coefficient and the code revision that produced it. Proposition 3 provides an executable correctness check: a run with a metric switched off must reproduce the parent method exactly, which we verify in §5 and Appendix I.

ETHICS STATEMENT

This work merges publicly available finetuned checkpoints and evaluates them on public benchmarks, using unlabeled examples from each task's own training split solely to estimate second-order statistics; no new data is collected, and no human subjects are involved. One merged Llama expert is a safety alignment expert, and Table 4 shows that every merging method we evaluate, including ours, recovers part of that expert's safety score. Merging is therefore not a safety-preserving operation by default, and a merged model should not inherit the safety claims of any parent without re-evaluation. Being a single static checkpoint in the base architecture, the merged model inherits the licenses and biases of the base and of every expert combined into it.

THE USE OF LARGE LANGUAGE MODELS

We used generative AI tools (LLM-based coding agents and chat assistants) throughout this project, and we report that use below against the categories of the ICLR policy. We have reviewed all AI-assisted work.

Tasks with required disclosure that we did use AI for. *Implementing methods.* The released library was written with substantial assistance from an LLM coding agent: the metric machinery (activation-statistics collection for C and the Jacobian-probe estimation of G, the whitened selection step and the pooled orthogonalization), the three evaluation pipelines (CLIP on the vision suites, RoBERTa on GLUE, Llama-3.2-3B on MergeBench), the sweep and caching infrastructure, and the scripts that emit the tables and figures of this paper. The baselines we re-implement rather than cite were ported with the same assistance. *Designing and giving feedback on experiments.* The shape of several ablations, the site-attribution grid of §5, the power sweep of §5, the sample-stability study of Appendix M and the metric-off correctness control implied by Proposition 3, was discussed with an LLM before being run, and in places proposed by one. *Refining hypotheses and the conceptual framing, and formulating mathematical claims.* The reading of $p = \frac{1}{2}$ as the power at which the orthogonalization step decorrelates tasks *as the data measures it*, the K-FAC reading of the two-sided metric, and the statement that the Euclidean method is the metric-off special case of ours were sharpened in dialogue with an LLM. *Interpreting results.* LLMs were used to summarize sweep outputs, to discuss possible explanations for results we did not expect, and to argue against those explanations. *Translation.* Parts of the manuscript were first drafted in the authors' main language and rendered into English with LLM assistance.

Tasks with required disclosure that we did not use AI for. No synthetic data was generated: every dataset is a public benchmark split used as distributed, and no dataset was cleaned or reformatted by an LLM beyond the loading code noted above. The proofs in Appendices F–I and the derivation of §4 were written and checked by the authors; where an LLM contributed to the framing, as stated above, it did not supply a step of a proof that we did not independently verify. Support for qualitative and thematic data analysis is not applicable to this work.

Tasks with recommended disclosure. We additionally used generative AI tools to draft and edit parts of this manuscript for readability, to produce the plotting code behind our figures, to suggest experimental parameters (coefficient grids, rank budgets and statistics budgets, all of which are then selected empirically on validation data, never by the model's suggestion), and for references literature search.

Verification. Every number reported in this paper is the output of a run of the released code under a released configuration; no result, table entry or score was produced by, or copied from, an LLM. Code that produces reported numbers was reviewed by the authors, and the method carries an executable check of its own: Proposition 3 requires a run with the metric switched off to reproduce the parent method, which we verify in §5 and Appendix I. All citations were checked against the cited papers, so that no reference or attributed claim rests on an LLM's output. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of AI.

A LIMITATIONS

Given the experts and the cached statistics, the merge is deterministic, so the uncertainty that matters is over *which* N unlabeled examples define C. We measure it directly: five independent draws (Appendix M) give a standard deviation of 0.11 normalized points for F-TSV on **ViT-B/32** at $T{=}8$ and 0.48 on GLUE, against which the pooled-metric orthogonalization's increment is 3.76 and 7.97 respectively, that is 33 and 14 standard deviations. We consider differences of that size safe, smaller differences are not, and we mark them where they occur. Two sources of uncertainty remain: the first is the statistics sample, i.e. *which* N unlabeled examples define C: Appendix M measures how far the metric moves when they are redrawn, on one backbone and for the input metric only. The second is the MergeBench evaluation, two of whose pillars are scored by sampled generation: the same weights returned five-pillar averages spanning 0.14 points across three evaluations (Appendix Q), which is the resolution at which the MergeBench rows should be read.

The margin over the strongest baseline is method-dependent. Against TSV-M the input metric is worth $+9.9$ normalized points on GLUE, but against RegMean, the strongest baseline there, it is $+2.4$ (Table 2). The two methods also swap ranks by modality: RegMean is competitive on GLUE and clearly behind on vision. We therefore claim a consistent improvement rather than a uniform.

The output metric underperforms. Paired across five draws of the statistics sample with the coefficient pinned at its published value, F-TSV^2 exceeds F-TSV by 0.15 ± 0.05 normalized points on **ViT-B/32** ($T{=}8$), positive in all five draws, and by 0.13 ± 0.76 on GLUE, where the per-draw difference ranges from -0.93 to $+0.93$ and changes sign. The vision margin is therefore real and small; the GLUE margin is not distinguishable from the draw. Allowing each draw to re-select its own coefficient, as the reported cells do, adds enough selector noise to bury the vision effect as well (0.12 ± 0.13). We report the output metric as completing the K-FAC picture at a cost in backward passes that grows with model size, and not as a result the paper rests on.

Coefficient selection on MergeBench. A generative benchmark offers no held-out accuracy at a per-α cost, so α is chosen there by teacher-forced validation likelihood (Appendix Q). The three rows of the MergeBench chain in Table 16 select $\alpha = 0.8$, 0.8 and 1.8, so its increments combine the change of geometry with a change of coefficient; we intentionally did not evaluate the chain at a common α, as it is done in model merging literature, and read that column as directionally consistent with vision and GLUE rather than as a measurement of the geometry's size.

Assumptions and scope. Assumption 2 treats input statistics and downstream Jacobians as independent, where that assumption does not hold, the two-sided norm misrepresents the curvature, and we treat this as a possible cause for the limited performance of the double-sided variant. As with all task-vector methods, we assume a single shared pretrained model.

B THEORY

We first state the assumptions underlying the derivation of the metric, which is well-defined for any PSD C_t and G_t:

Assumption 1 (Local quadratic losses)**.** *$\mathcal{L}_t(\theta) \approx \mathcal{L}_t(\theta_t) + \frac{1}{2}(\theta - \theta_t)^\top H_t(\theta - \theta_t)$ about each expert, with $H_t \succeq 0$ (L is reserved for the merged left factor).*

Assumption 2 (K-FAC factorization)**.** *Per layer, $\mathbb{E}[(xx^\top) \otimes (J^\top J)] \approx C_t \otimes G_t$ is exact whenever J is constant over the data (Martens & Grosse, 2015).*

Assumption 3 (Spectral sparsity)**.** $\Delta = \sum_t \Delta_t^{(k)}$ *with* $\operatorname{rank}(\Delta_t^{(k)}) \le k = \lfloor \min(m, n)/T \rfloor$.

Table 5: **Notation**. Merging acts per weight matrix, layer indices are removed to simplify notation. The input metric is C and the output metric is G at every stage, raw or conditioned, and a metric enters the merge only through a power p of itself. Decorations are used consistently: $\bar{\cdot}$ pools over tasks, $\hat{\cdot}$ stacks per-task factors, $\tilde{\cdot}$ lives in whitened coordinates, $\cdot^{(k)}$ is a task's rank-k block.

$\theta_0,\ \theta_t$	base weights; expert t	$C_t \in \mathbb{R}^{n\times n}$	conditioned $\mathbb{E}_{x\sim\mathcal{D}_t}[xx^\top]$
$\tau_t = \theta_t - \theta_0$	task vector of expert t	$G_t \in \mathbb{R}^{m\times m}$	conditioned $\mathbb{E}[J^\top J]$
$\Delta_t \in \mathbb{R}^{m\times n}$	one weight matrix of τ_t	$\bar{C}, \bar{G}$	pooled metrics, $\bar{C} = \frac{1}{T}\sum_t \mathrm{unit}(C_t)$
$\Delta_t^{(k)} = U_t^{(k)} S_t^{(k)} V_t^{(k)\top}$	task t's rank-k block	$C_t^p,\ G_t^{p_g}$	whitening factors (powers of a metric)
$\Delta,\ \Delta^\star$	merged update; $\star$ marks the solution	$p,\ p_g$	whitening powers, derived $p = p_g = \frac{1}{2}$
$\theta = \theta_0 + \alpha\Delta^\star$	the merged model	$r,\ r_g$	conditioning ranks, absolute or a fraction of the width
$\hat{U}, \hat{s}, \hat{V}$	per-task factors, stacked	$\mathrm{unit}(\cdot)$	rescaling to unit mean eigenvalue
$L,\ R$	the stacks, orthogonalized	$\mathrm{cond}(\cdot\,; r)$	the conditioning map of Appendix E
$T,\ k$	tasks; rank kept per task and layer	$\kappa(\cdot)$	condition number of a conditioned metric
N	unlabeled examples per task	n_p	Gaussian VJP probes per batch
α	merging coefficient	$\mathrm{polar}(\cdot)$	nearest orthonormal factor

Under Assumptions 1 to 3 the per-layer term of equation 2 is the two-sided functional norm (Appendix D).

Proposition 1 (Nearest metric-orthonormal factor)**.** *Let $A \succ 0$ be a conditioned metric and $p > 0$. Then $R^\star = \mathrm{polar}(\hat{V}^\top A^{\,p})A^{-p}$ minimizes $\|(R - \hat{V}^\top)A^{\,p}\|_F$ subject to $R\,A^{\,2p}R^\top = I$, and attains it. Two corners: at $p = \frac{1}{2}$ the constraint is A-orthonormality, the non-interference criterion of equation 7; at $A = I$ the solution is the Euclidean polar factor. Taking $A = \bar{C}$ gives equation 8, and the mirror statement for columns orthonormal in $\bar{G}$ gives equation 9.*

Proposition 2 (The orthogonalization fixes the power)**.** *Let $\bar{C} \succ 0$ be the conditioned pooled metric with $\bar{C} \neq I$, and write $\|M\|_{\bar{C}}^2 := \mathrm{tr}(M\bar{C}M^\top)$ for the quadratic form of equation 7, which equals $\mathbb{E}_x\|Mx\|^2$ when $\bar{C}$ is the second moment of x. Then (i) the constraint $R\,\bar{C}^{\,2p}R^\top = I$ of equation 8 defines the same set of matrices as $\bar{C}$-orthonormality, $R\,\bar{C}\,R^\top = I$, if and only if $p = \frac{1}{2}$; and (ii) the objective $\|(R - \hat{V}^\top)\bar{C}^{\,p}\|_F^2$ equals $\|R - \hat{V}^\top\|_{\bar{C}}^2$ for every R if and only if $p = \frac{1}{2}$. The mirror statements hold on the output side with $\bar{G}$ and p_g.*

Selection, by contrast, uses C_t^p only to rank directions and never rescales the kept block equation 6; it is well defined at every $p > 0$ and constrains none. The power is therefore fixed by the orthogonalization alone, the step that Figure 2 finds to carry the gain, and §5 measures it there.

Lemma 1 (Range alignment)**.** *With $G_t^{p_g}\Delta_t C_t^p = \tilde{U}\tilde{S}\tilde{V}^\top$ and $Q_t = \mathrm{QR}(C_t^p\tilde{V}_k)$, $\mathrm{range}(\Delta_t Q_t) = \mathrm{range}(G_t^{-p_g}\tilde{U}_k)$: selecting the top-$k$* input *directions of the doubly whitened update confines the block's outputs to the top-k whitened* output *subspace, so no separate left truncation is needed (Appendix H).*

Proposition 3 (Nested exactness)**.** *Per layer: (i) if $r_g = 0$, $p_g = 0$, or no output statistic was observed,* F-TSV^2 *coincides with* F-TSV*; (ii) if in addition $r = 0$, $p = 0$, or no input statistic was observed, it coincides with TSV-M, i.e. with equation 1. In every case the switched-off metric enters as the exact identity, $A^{\,p} = A^{-p} = I$, so the reductions are identities of the computation up to floating-point roundoff, not limits: the Euclidean method is a corner of the family along either axis, rank or power.*

Proofs are in Appendices F–I; Appendix I also verifies the reductions numerically. Proposition 3 makes every ablation cell of §5 an interpolation between two published methods: a site that keeps no metric executes, at that layer, exactly the TSV-M method.

C JLS APPROXIMATION LADDER

Algorithm 1 states the two methods. Table 6 places them among the instances of equation 2. Each row fixes the two ingredients of the objective: the curvature H_t that prices a deviation from expert t, and the constraint Ω on the merged update. The first four rows impose no constraint and solve equation 2 exactly, in closed form or, for MaTS, by conjugate gradients. They differ only in how

Algorithm 1 Functional TSV (F-TSV) and its two-sided extension (F-TSV^2). Dropping every G_t gives F-TSV; dropping every C_t as well gives TSV-M (Proposition 3).

Require: base θ_0; experts $\theta_1..\theta_T$; N unlabeled examples per task; conditioning rank r and power p on the input side and, two-sided, (r_g, p_g, n_p) on the output side; ranks are absolute or a fraction of the layer width

Statistics. For each task t: accumulate C_t over hooked layer inputs (token rows, padding masked); two-sided, accumulate G_t from n_p Gaussian VJP probes per batch; on a causal LM, the two factors of the token-embedding matrix (Appendix L). Cache to disk.
for all weight matrices $W \in \mathbb{R}^{m\times n}$ **do**
  $k \leftarrow \max(1, \lfloor \min(m,n)/T \rfloor)$
  $\bar{C} \leftarrow \mathrm{cond}(\frac{1}{T}\sum_t \mathrm{unit}(C_t); r)$, $\bar{G} \leftarrow \mathrm{cond}(\frac{1}{T}\sum_t \mathrm{unit}(G_t); r_g)$ ▷ pooled from the raw statistics
  for all tasks $t = 1..T$ **do**
    $C_t \leftarrow \mathrm{cond}(C_t; r)$ if C_t observed, else I; likewise $G_t \leftarrow \mathrm{cond}(G_t; r_g)$
    if either metric is active **then**
      $(\cdot,\cdot,\tilde{V}) \leftarrow \mathrm{SVD}(G_t^{p_g}\Delta_t C_t^{p})$; $Q_t \leftarrow \mathrm{QR}(C_t^{p}\tilde{V}_{[:,1:k]})$ ▷ the metric selects the subspace
      $(U_t^{(k)}, S_t^{(k)}, V_t^{(k)}) \leftarrow$ SVD of $\Delta_t Q_t Q_t^\top$ ▷ content undistorted, equation 6
    else
      $(U_t^{(k)}, S_t^{(k)}, V_t^{(k)}) \leftarrow$ top-k SVD of Δ_t ▷ equal to TSV-M
    end if
  end for
  stack the blocks into $\hat{U}, \hat{s}, \hat{V}^\top$
  $R \leftarrow \mathrm{polar}(\hat{V}^\top \bar{C}^{\,p})\,\bar{C}^{\,-p}$ if C active else $\mathrm{polar}(\hat{V}^\top)$ ▷ equation 8
  $L \leftarrow \bar{G}^{\,-p_g}\,\mathrm{polar}(\bar{G}^{\,p_g}\hat{U})$ if G active else $\mathrm{polar}(\hat{U})$ ▷ equation 9
  merged layer $\leftarrow L\,\mathrm{diag}(\hat{s})R$
end for
non-matrix parameters $\leftarrow$ mean of task vectors
return $\theta_0 + \alpha\Delta^\star$, α selected on validation data

much structure H_t keeps, from the identity through the diagonal Fisher to Kronecker factors of a layer's input statistics (RegMean) and of both of its sides (MaTS). With the coefficient α of §3, the first row is task arithmetic, and $\alpha = 1$ gives weight averaging. The exact solutions of the Kronecker rows multiply by the inverse of a sum of estimated statistics, $(\sum_t C_t)^{-1}$ for RegMean, and keep no per-task low-rank structure. The last three rows replace $\Omega = 0$ by the per-task spectral constraint of Assumption 3, and each keeps the curvature of an unconstrained row: TSV-M the identity, F-TSV the $C_t \otimes I$ of RegMean, and F-TSV^2 the two-sided Kronecker form of MaTS, with the label-free sensitivity $G_t = \mathbb{E}[J^\top J]$ of §4.2 as its output factor.

Table 6: Merging methods as instances of equation 2. Kronecker factors are stated for column-stacked $\mathrm{vec}(\Delta)$, where $\mathrm{vec}(\Delta)^\top (C \otimes G)\,\mathrm{vec}(\Delta) = \|G^{1/2}\Delta C^{1/2}\|_F^2$ (Appendix D). The last two rows are this paper.

H_t	Ω	Solution of equation 2	Method
I	0	$\Delta = \frac{1}{T}\sum_t \tau_t$	Task Arithmetic
$\mathrm{diag}(F_t)$	0	per-coordinate weighted mean	Fisher merging
$C_t \otimes I$ (per layer)	0	$\Delta = (\sum_t \Delta_t C_t)(\sum_t C_t)^{-1}$	RegMean
block K-FAC	0	CG solve of the normal equations	MaTS
I	spectral, per task	select k dirs/task + orthogonalize	TSV-M
$C_t \otimes I$	spectral, per task	§4.1	F-TSV (ours)
$C_t \otimes G_t$	spectral, per task	§4.2	F-TSV^2 (ours)

D THE FUNCTIONAL NORM AND ITS KRONECKER FORM

One-sided identity. For $\Delta \in \mathbb{R}^{m\times n}$ and $x \sim \mathcal{D}$ with $C = \mathbb{E}[xx^\top]$,

$$\mathbb{E}\|\Delta x\|^2 = \mathbb{E}\big[\operatorname{tr}(\Delta^\top \Delta\, xx^\top)\big] = \operatorname{tr}(\Delta^\top \Delta C) = \operatorname{tr}(C^{1/2}\Delta^\top \Delta C^{1/2}) = \|\Delta C^{1/2}\|_F^2.$$

The functional size of an update is thus its Frobenius norm after right-whitening: input directions with large data energy are stretched, directions the data never visits are shrunk.

Two-sided approximation. With $J(x) = \partial f/\partial y$ the Jacobian of the final representation with respect to the layer output, a perturbation $\delta y = \Delta x$ changes f by $J(x)\Delta x$ to first order, so

$$\mathbb{E}\|J(x)\Delta x\|^2 = \mathbb{E}\big[\operatorname{tr}(\Delta^\top J^\top J\, \Delta\, xx^\top)\big] \;\approx\; \operatorname{tr}(\Delta^\top G \Delta C) = \|G^{1/2}\Delta C^{1/2}\|_F^2,$$

where the approximation is Assumption 2, exact whenever J is constant over the data.

Kronecker form. For column-stacked vectorization, $(A \otimes B)\operatorname{vec}(X) = \operatorname{vec}(BXA^\top)$, so for symmetric C, G,

$$\operatorname{vec}(\Delta)^\top (C \otimes G)\operatorname{vec}(\Delta) = \operatorname{vec}(\Delta)^\top \operatorname{vec}(G\Delta C) = \operatorname{tr}(\Delta^\top G \Delta C) = \|G^{1/2}\Delta C^{1/2}\|_F^2.$$

For $y = Wx$ we have $y = (x^\top \otimes I_m)\operatorname{vec}(W)$, hence the Gauss–Newton block restricted to $\operatorname{vec}(W)$ is $\mathbb{E}[(xx^\top) \otimes (J^\top J)] \approx C \otimes G$, whose quadratic form is the two-sided norm above.

E METRIC CONDITIONING

Estimation. For every observable layer, the N unlabeled examples of task t are passed once through the pretrained model θ_0, and the second moment of the layer's input is accumulated over the M token rows they produce, with padding masked: $C_t = \frac{1}{M}\sum_{i=1}^{M} x_i x_i^\top$. On vision and GLUE $N = 512$; Table 15 lists every budget. Because the estimate comes from a small sample, every metric is conditioned before the merge uses it.

The conditioning map. Rescale the statistic to unit mean eigenvalue and write its eigendecomposition as $\sum_j \lambda_j w_j w_j^\top$, with $\frac{1}{n}\sum_j \lambda_j = 1$. We keep the top r eigendirections and floor the remaining tail to the isotropic level,

$$C_t \;\propto\; I \;+\; \sum_{j\le r} (\lambda_j - 1)\, w_j w_j^\top, \tag{10}$$

and rescale the result to unit mean eigenvalue. Conditioning moves eigenvalues only, and we keep calling the result C_t throughout the paper. The same map, $\operatorname{cond}(\cdot\,; r)$ in Algorithm 1, conditions every metric the method uses, per-task or pooled, input or output: C_t, $\bar{C}$, G_t and $\bar{G}$. Its only hyperparameter is the rank r, absolute on the vision and GLUE encoders and a fraction of each layer's width on **Llama-3.2-3B** (Table 15). Given a raw statistic $A \in \mathbb{R}^{n\times n}$, a rank r and a power $p > 0$, the map

1. symmetrizes, $A \leftarrow \frac{1}{2}(A + A^\top)$, and returns identity factors if $\operatorname{tr} A \le 0$, a degenerate statistic;
2. rescales to unit mean eigenvalue, $A \leftarrow A \cdot n/\operatorname{tr}(A)$;
3. eigendecomposes $A = U\Lambda U^\top$ once, keeps the top r eigenvalues, sets the other $n - r$ to 1, and rescales to unit mean eigenvalue;
4. returns the whitening factor $A^{\,p} = U\operatorname{diag}(\lambda^p)U^\top$ and, where needed, its inverse $A^{\,-p} = U\operatorname{diag}(\lambda^{-p})U^\top$.

Every power follows from the single eigendecomposition of step 3, and for $r \ll n$ each factor is a rank-r update of a scaled identity.

Why the tail is floored. The merge uses a metric only through its powers. Truncating the tail to zero would leave $A^{\,-p}$ undefined; flooring it at the isotropic level keeps the tail's mass and every eigenvalue away from zero. This bounds the only inverse factors the merge forms, $\bar{C}^{\,-p}$ in equation 8 and, for F-TSV^2, $\bar{G}^{\,-p_g}$ in equation 9: with unit mean eigenvalue, each stretches a direction by at most κ^p, with κ the condition number of the conditioned metric. The per-task metrics C_t and G_t enter only through positive powers, in the selection step, and are never inverted. At $r = 0$ every

eigenvalue is 1 and $A^{p} = A^{-p} = I$ exactly: this is the TSV-M anchor of Proposition 3, reached along the same axis r that controls the conditioning. The power axis reaches the same anchor at $p = 0$, where the factors are also returned as the exact identity.

Measured condition numbers. On **ViT-B/32** (T=8), over the 48 observable matrices, the conditioned $\bar{C}$ has a median κ of 260 (range 90 to 1456), so the amplification at $p = \frac{1}{2}$ is 16 in the median and at most 38. When the statistics sample is redrawn, κ moves by 0.35% in the median and by at most 4.1% (Appendix M).

The rank. Sweeping $r \in \{0, 32, 64, 128, 256, 512\}$ with everything else fixed, the normalized test accuracy is 92.04, 95.55, 95.63, 96.02, 95.70 and 95.93 on **ViT-B/32** (T=8), and 78.02, 85.95, 85.61, 87.86, 85.47 and 85.87 on GLUE. At $r = 32$ the metric already delivers 88% (vision) and 80% (GLUE) of the gain over $r = 0$ that $r = 128$ delivers. The test score peaks at $r = 128$ on both benchmarks and is not monotone above it. On the validation split, which selects each cell's α, the five ranks from 32 to 512 lie within 0.15 normalized points of one another on **ViT-B/32**, highest at $r = 32$; on GLUE, validation also peaks at $r = 128$. At $r = 0$ the metric is the identity, and the sweep reproduces TSV-M.

F Proof of Proposition 1

Euclidean case. $\mathrm{polar}(Z) = \tilde{U}\tilde{V}^{\top}$ from $Z = \tilde{U}\tilde{S}\tilde{V}^{\top}$ minimizes $\|Z - X\|_F$ over X with orthonormal rows (equivalently columns).

Metric case. We seek R nearest to $\hat{V}^{\top}$ in the functional norm subject to $R\,\bar{C}^{\,2p}R^{\top} = I$: $\min_R \|(R - \hat{V}^{\top})\bar{C}^{\,p}\|_F$. Substituting $\tilde{R} = R\,\bar{C}^{\,p}$ turns the constraint into $\tilde{R}\tilde{R}^{\top} = I$ and the objective into $\|\tilde{R} - \hat{V}^{\top}\bar{C}^{\,p}\|_F$, whose minimizer is $\tilde{R} = \mathrm{polar}(\hat{V}^{\top}\bar{C}^{\,p})$ by the Euclidean case. Mapping back gives $R = \mathrm{polar}(\hat{V}^{\top}\bar{C}^{\,p})\bar{C}^{\,-p}$ and $R\,\bar{C}^{\,2p}R^{\top} = \tilde{R}\tilde{R}^{\top} = I$, which is equation 8; at $p = \frac{1}{2}$ the constraint reads $R\,\bar{C}\,R^{\top} = I$. The output-side rule equation 9 follows from the mirror substitution $\tilde{L} = \bar{G}^{\,p_g}L$. At $\bar{C} = I$ (equivalently $p \to 0$) both reduce to the Euclidean polar factor. □

Remark 1. *A row of R has unit* data *norm, so along directions the data barely excites its Euclidean length may exceed one, intentionally, since weight mass there is functionally cheap. The isotropic tail caps the effect at $\kappa(\bar{C})^p$ (Appendix E), and §5 is where that trade stops being free.*

G Proof of Proposition 2

Both statements reduce to one fact: two symmetric positive definite matrices with the same quadratic form are equal. Write $A = \bar{C}^{\,2p}$ and $B = \bar{C}$.

Statement (ii). $\|M\bar{C}^{\,p}\|_F^2 = \mathrm{tr}(MAM^{\top})$ and $\|M\|_{\bar{C}}^2 = \mathrm{tr}(MBM^{\top})$. If the two agree for every M, taking $M = w^{\top}$ for an arbitrary vector w gives $w^{\top}Aw = w^{\top}Bw$ for all w, hence $A = B$; the converse is immediate.

Statement (i). Suppose every $R \in \mathbb{R}^{q\times n}$ ($q \le n$) with $RAR^{\top} = I$ also satisfies $RBR^{\top} = I$. Take any w with $w^{\top}Aw = 1$, extend it to q vectors that are A-orthonormal, possible since $q \le n$, and stack them as the rows of R. Then $RAR^{\top} = I$, so $RBR^{\top} = I$, so $w^{\top}Bw = 1$: every A-unit vector is a B-unit vector, and by scaling $w^{\top}Aw = w^{\top}Bw$ for all w, hence $A = B$. The converse is immediate.

$A = B$ forces $p = \frac{1}{2}$. $\bar{C}$ is symmetric with eigenvalues $\lambda_j > 0$, and $\bar{C}^{\,2p}$ has the same eigenvectors with eigenvalues λ_j^{2p}, so $\bar{C}^{\,2p} = \bar{C}$ iff $\lambda_j^{2p} = \lambda_j$ for every j. Since $\bar{C} \neq I$ some $\lambda_j \neq 1$, and for that j the equation holds iff $2p = 1$.

Output side. Substitute $\tilde{L} = \bar{G}^{\,p_g}L$ as in Appendix F and use $\|u\|_{\bar{G}}^2 = u^{\top}\bar{G}u$, the quadratic form of $\langle u, u'\rangle_{\bar{G}}$ in §4.2; the two statements become their mirror images and the same argument gives $p_g = \frac{1}{2}$. □

Remark 2. *When $\bar{C} = I$ (conditioning rank $r = 0$, Appendix E) every p gives the same Euclidean polar factor and nothing is fixed. The power is determined exactly when there is a data geometry to determine it against, which is the Euclidean anchor of Proposition 3 seen from the other side.*

H PROOF OF LEMMA 1

Let $G_t^{p_g}\Delta_t C_t^p = \tilde{U}\tilde{S}\tilde{V}^\top$ and $Q_t = \mathrm{QR}(C_t^p\tilde{V}_k)$, so $\mathrm{range}(Q_t) = \mathrm{range}(C_t^p\tilde{V}_k)$. Then

$$\Delta_t\, C_t^p\tilde{V}_k = G_t^{-p_g}\big(G_t^{p_g}\Delta_t C_t^p\big)\tilde{V}_k = G_t^{-p_g}\tilde{U}\tilde{S}\tilde{V}^\top\tilde{V}_k = G_t^{-p_g}\tilde{U}_k\tilde{S}_k,$$

hence $\mathrm{range}(\Delta_t Q_t) = \mathrm{range}(G_t^{-p_g}\tilde{U}_k)$: keeping the top-$k$ *input* directions of the doubly whitened update confines the block's outputs to the top-k whitened output subspace mapped back to raw coordinates. Selecting on one side selects both. □

I PROOF OF PROPOSITION 3

At $r = 0$ the conditioned metric is the identity exactly (Appendix E): no eigendirection is retained, every eigenvalue sits at the isotropic level, and $C_t^p = C_t^{-p} = I$ for every p; at $p = 0$ the factors are the identity for every conditioned metric; and a layer whose statistic was never observed is assigned the identity. With the output side off, every G factor in Algorithm 1 is a multiplication by I, so each layer executes the one-sided computation. With both sides off, the whitened update is Δ_t itself, Q_t spans its top-k right singular subspace, $\Delta_t Q_t Q_t^\top$ is its rank-k truncation, and both polar steps are Euclidean, which is equation 1. The two computations differ only by the extra factorizations of the metric path (a QR of an already orthonormal basis and a second SVD of the same block), which agree in exact arithmetic. □

Verification. On the reported cells, running F-TSV^2 with the output metric kept at no site returns 86.81 (96.02) on **ViT-B/32** (T=8) and 73.13 (87.86) on GLUE, the F-TSV rows of Tables 1 and 2 in every stored digit (the none rows of the output panels of Tables 3 and 14). On the aggregation itself, in double precision on synthetic low-rank task vectors (five tasks, three layers of widths 64 to 128), the three routes to the identity input metric, $p = 0$, $r = 0$, and no observed statistic, agree with the plain TSV-M aggregation to a relative difference below 2×10^{-14}. On MergeBench, where the evaluation is not deterministic (Appendix Q), the p=0 cell of Table 13 (39.37) and the TSV-M row of Table 4 (39.45) are two evaluations of the same weights.

J F-TSV PRESERVES THE EXPERT UPDATE

The abstract states that F-TSV uses the data metric to choose each task's subspace and to define cross-task orthogonality, while preserving the expert update within the chosen subspace. This appendix states what is preserved, by which step, and what the orthogonalization step changes.

Step 1 keeps the expert's own action. Selection uses the metric only to choose the orthonormal basis Q_t of equation 6, and the kept block is $\Delta_t^{(k)} = \Delta_t Q_t Q_t^\top$. Hence

$$\Delta_t^{(k)}x = \Delta_t x \;\text{ for every } x \in \mathrm{range}(Q_t), \qquad \Delta_t^{(k)}x = 0 \;\text{ for every } x \perp \mathrm{range}(Q_t)\,:$$

on the input directions it keeps, the block responds exactly as the expert does. No power of C_t multiplies the kept content, and the singular values $\hat{s}_t$ of the block are those of $\Delta_t Q_t$, the expert restricted to the chosen subspace, and it holds for every power p and every conditioning rank r.

Step 2 keeps the singular values and the subspaces, and renormalizes the directions. The pooled orthogonalization equation 8 acts on the stacked factors $\hat{U}, \hat{s}, \hat{V}$ of the T blocks. Since $\mathrm{polar}(Z) = (ZZ^\top)^{-1/2}Z$ whenever Z has full row rank, as $Z = \hat{V}^\top\bar{C}^{\,p}$ does,

$$R = \mathrm{polar}\left(\hat{V}^\top\bar{C}^{\,p}\right)\bar{C}^{\,-p} = \left(\hat{V}^\top\bar{C}^{\,2p}\hat{V}\right)^{-1/2}\hat{V}^\top,$$

whereas TSV-M uses

$$\mathrm{polar}(\hat{V}^\top) = \left(\hat{V}^\top\hat{V}\right)^{-1/2}\hat{V}^\top.$$

Both are the symmetric orthogonalization of the selected directions; F-TSV computes their Gram matrix in the data metric instead of the Euclidean one. Three properties of the merged update $\Delta^\star = L\,\mathrm{diag}(\hat{s})R$, with $L = \mathrm{polar}(\hat{U})$, follow.

Table 7: The correspondence between functional maps and model merging for functional TSV.

Functional maps (Ovsjanikov et al., 2012)	**This manuscript**
A correspondence is a linear map between *function spaces*, not a point-to-point map.	A task update is a linear operator between *feature spaces*, not a list of weights.
The map is written in a chosen basis with a geometry-aware (area-weighted) inner product.	The operator is analyzed in the data-induced inner products $\langle v, v' \rangle_C$ and $\langle u, u' \rangle_G$.
The basis is truncated to few low-frequency eigenfunctions.	The operator is truncated to the top-k directions of $G^{p_g} \Delta_t C^p$.
Maps from isometries are orthonormal *in the function inner product*.	The retained directions are made orthonormal in $\bar{C}$ and $\bar{G}$.
The map is recovered by least squares preserving descriptors.	The merge is measured by the functional error $\sum_t \mathrm{tr}[(\Delta^\star - \Delta_t) C_t (\Delta^\star - \Delta_t)^\top]$ of equation 2.
The point map is recovered from the functional map.	The merged weights are recovered as $\theta_0 + \alpha \Delta^\star$.

- *It stays in the chosen subspaces.* Every row of R is a combination of the selected directions, so $\Delta^\star$ reads its input only from $\mathrm{range}(\hat{V})$, the span of the directions selected in Step 1, and writes only into $\mathrm{range}(\hat{U})$, as in TSV-M.

- *It keeps the singular values $\hat{s}$.* $\Delta^\star \bar{C}^p = L \, \mathrm{diag}(\hat{s}) \, \mathrm{polar}(\hat{V}^\top \bar{C}^p)$ is $\mathrm{diag}(\hat{s})$ between two orthonormal factors, so $\hat{s}$ are exactly the singular values of the merged update in the whitened coordinates of the data metric, where TSV-M has them in weight coordinates. On the merged layers of all nine vision settings and GLUE, $\|\Delta^\star \bar{C}^{1/2}\|_F / \|\hat{s}\|_2 = 1.000$ (median over layers).

- *It changes the Euclidean length of the directions.* Each row r of R has unit length in the data metric, $\|r \, \bar{C}^p\| = 1$, so its Euclidean length lies between $\lambda_{\max}(\bar{C})^{-p}$ and $\lambda_{\min}(\bar{C})^{-p}$. Since $\bar{C}$ has unit mean eigenvalue, directions that the pooled data excites more than the average are shortened and those it excites less are lengthened, by a factor of at most $\kappa(\bar{C})^p$, which is 16 in the median layer of **ViT-B/32** and at most 38 (Appendix E). This is what orthonormality in the data metric means (the remark in Appendix F), and it acts even on a single task: at $T = 1$, on a layer with $n \leq m$, TSV-M returns Δ_1 whereas F-TSV returns $\Delta_1 \bar{C}^{-p}$.

The overall change of scale is absorbed by α. Because $\hat{s}$ is carried in the data metric rather than in weight coordinates, the merged update of F-TSV is smaller than that of TSV-M on the data, though not in weight space. Per layer, its size on the pooled inputs, $\mathrm{tr}(\Delta^\star C_{\text{mix}} \Delta^{\star\top})^{1/2}$ with $C_{\text{mix}} = \frac{1}{T} \sum_t C_t$, is 0.49 to 0.58 times that of TSV-M (median over layers, nine vision settings and GLUE), while its Frobenius norm is 1.01 to 1.26 times as large. The validated coefficients compensate: $\alpha_{\text{TSV-M}} / \alpha_{\text{F-TSV}}$ is 0.47 to 0.60, within 0.08 of the size ratio in every setting. This is why the α of F-TSV in Table 16 is about twice that of TSV-M, while that of metric-guided selection alone is not. Run at the coefficient that gives it the step of F-TSV on the data, TSV-M scores as at its own coefficient: 92.04, 85.64 and 78.57 normalized on **ViT-B/32** at T=8 and T=20 and on GLUE, against 92.05, 85.91 and 78.01. At their validated coefficients the two methods therefore differ in which directions they keep and in how the kept directions are normalized and combined, not in the size of the step they take on the data.

Summary. F-TSV keeps the expert's own action on the subspace that the metric chooses, with its own singular values, and the metric never multiplies that content. The orthogonalization then keeps those singular values and those subspaces, and makes the retained directions orthonormal in the data metric instead of the Euclidean one. The merged update therefore carries the kept singular values $\hat{s}$ as its singular values in the data metric rather than in weight coordinates, and the resulting change of overall scale is absorbed by α.

K THE FUNCTIONAL-MAPS DICTIONARY

Functional maps (Ovsjanikov et al., 2012; 2016) represent a correspondence between two shapes by the linear operator it induces between their function spaces. That operator is written in truncated eigenbases that are orthonormal under each shape's area-weighted inner product $\langle f, g\rangle = f^\top A g$, with A the mass matrix of the mesh. Table 7 lists the counterpart of each ingredient in this paper. The task update Δ_t is the operator, weight coordinates play the role of mesh vertices, and the two mass matrices become the two metrics of a layer: C on its inputs, with $\langle \Delta, \Delta'\rangle_C = \mathbb{E}_x\langle \Delta x, \Delta' x\rangle = \mathrm{tr}(\Delta C \Delta'^\top)$, and G on its outputs. Setting both metrics to the identity recovers TSV-M (Proposition 3); it corresponds to setting the mass matrices to the identity, so that updates are measured by their weight coordinates alone. At the derived powers $p = p_g = \frac{1}{2}$, truncation (Step 1) and orthonormality (Step 2) are defined in these inner products, as they are for the eigenbasis of a functional map. The correspondence is structural: the eigenbasis of a functional map is fixed by the shape, whereas the directions retained here are computed per task from the whitened update.

L THE TOKEN-EMBEDDING METRIC OF A CAUSAL LANGUAGE MODEL

A causal LM merges one large matrix that no forward hook on a linear layer can observe: the token embedding $U \in \mathbb{R}^{|\mathcal{V}|\times n}$, tied to the output head, 128256×3072 on **Llama-3.2-3B**. It holds 12.3% of the parameters and, measured on the five MergeBench experts, between 13.6% (safety) and 66.0% (multilingual) of each expert's task-vector energy $\|\tau_t\|^2$ (42.0% math, 17.1% coding, 14.2% instruction). Its input is nonetheless well defined. Read as the lookup $h_0 = U^\top e_v$, with e_v the one-hot token, a perturbation Δ of U changes the final representation by $J\,\Delta^\top e_v$ to first order, and the functional norm of equation 5 becomes

$$\begin{aligned} \mathbb{E}_v\|J\,\Delta^\top e_v\|^2 = \mathrm{tr}\big(\Delta\, G_h\, \Delta^\top F\big) = \big\|F^{1/2}\,\Delta\, G_h^{1/2}\big\|_F^2, \\ F = \mathrm{diag}(\text{token frequency}), \quad G_h = \mathbb{E}[J^\top J], \end{aligned} \tag{11}$$

so the token frequency is the metric on the vocabulary side and the downstream sensitivity at the embedding output, estimated with the same Gaussian VJP probes as G_t in §4.2, is the metric on the hidden side. In the stored $|\mathcal{V}| \times n$ layout, G_h occupies the right factor of equation 4 and F the left one: F-TSV, which consumes the right factor only, applies the hidden-side sensitivity, and F-TSV^2 applies both. The vocabulary-side factor is diagonal, which is what makes the layer tractable: a dense one would be 61.3 GiB before any eigendecomposition, whereas the diagonal costs 0.5 MB, and every step of Appendix E acts on eigenvalues only, so the machinery is unchanged. On this matrix the whole merge, five exact SVDs and both polar factors, takes 5.2 s at 15.3 GiB peak memory. The statistics are estimated from 128×2 held-out sequences per domain with 8 probes each (Table 15). The metric is active in every MergeBench row of this paper and does not exist on the encoder pipelines.

Table 8 repeats the chain of Figure 2, extended by F-TSV^2, with the embedding merged in the Euclidean metric and in the metric of equation 11, on all five pillars. With the embedding outside the metric, metric-guided selection is worth $+0.64$ on the five-pillar average and the pooled orthogonalization -1.87; with it inside, the two increments are -0.85 and $+1.20$, the signs reported in Figure 2. The output metric changes in the same way, from -0.86 against the one-sided method to -0.07. On a causal LLM, the pooled-metric orthogonalization therefore requires the token-embedding matrix to be inside the metric it orthogonalizes in.

M SAMPLE-LEVEL STABILITY OF THE DATA METRIC

The method reaches the data through one object per layer, the input second moment C, estimated from $N = 512$ unlabeled examples per task. Everything downstream, which subspace each expert keeps and what cross-task orthogonality means, is a function of C, so the uncertainty that matters is over *which* 512 examples define it. We measure it first on the metric itself, without merging or evaluating, and then on the accuracy of the merge.

Setup. On **ViT-B/32** with the 8-task suite, at the setting of the reported results ($r = 128$, $p = \frac{1}{2}$, base encoder, training split), we collect three independent draws (seeds 0, 1, 2) of every per-task statistic,

Table 8: **The token-embedding metric on MergeBench** (**Llama-3.2-3B**, five experts, all five pillars, $p{=}\frac{1}{2}$, each row at its own validation-selected α). The chain of Figure 2 plus F-TSV^2, with the tied token embedding merged in the Euclidean metric and in the metric of equation 11. TSV-M carries no metric anywhere and is common to both blocks. Differences below $\sim$0.2 on the average are within the evaluation spread of Appendix Q. F-TSV correspond to Δ_{ort}.

Embedding metric	Row	Instr.	Math	Coding	Multiling.	Safety	**Avg.**	α
	TSV-M	20.20	50.42	40.41	41.87	44.34	39.45	0.8
Euclidean	Δ_{sel}	21.63	50.87	41.27	41.73	44.95	40.09	0.9
	F-TSV	19.68	51.78	38.47	40.47	40.66	38.21	1.6
	F-TSV^2	18.05	51.93	36.12	40.50	40.15	37.35	1.5
Eq. equation 11	Δ_{sel}	18.81	49.05	39.60	41.80	43.72	38.60	0.8
	F-TSV	21.14	57.01	36.46	40.30	44.10	39.80	1.8
	F-TSV^2	20.53	56.33	34.53	40.41	46.84	39.73	1.8

together with draws of $N \in \{128, 256, 1024\}$, class-balanced draws, draws from the validation and test splits, draws through the finetuned experts, and two disjoint halves of one permutation: 27 draws in all, compared layer by layer. The distance between two estimates A, B of the same layer's metric is $\|A - B\|_F / \operatorname{mean}(\|A\|_F, \|B\|_F)$ on unit-scaled metrics, taken at the three stages the merge puts C through: raw, conditioned, and the whitening factor C^p the merge multiplies by. The agreement of two kept top-r eigenspaces is $\|U_A^\top U_B\|_F^2 / r$, reported as a multiple of its chance level r/n, which is 0.167 at width 768 and 0.042 at width 3072. Every number is a median over the 48 observable matrices, times the 8 tasks for per-task metrics, and is read against the distance between two *different tasks'* metrics from the same draw: the scale on which a distance between two estimates of the same quantity is small or large.

Redrawing the sample is a second-order effect. Two draws of 512 examples differ by 0.064 on the raw metric and 0.065 on C^p, against 1.139 and 0.547 between two different tasks: $17.7\times$ and $8.4\times$ less (Table 9, Figure 3a). The kept top-r eigenspace of a redraw agrees at $5.6\times$ chance, against $3.3\times$ for two different tasks.

Table 9: **Two draws of the same statistics sample** (**ViT-B/32**, $T{=}8$, $r{=}128$, $p{=}\frac{1}{2}$). Median relative distance between two independent draws of $N{=}512$ examples over the 48 observable matrices ($\times 8$ tasks for per-task metrics), at the three stages of Appendix E, with the interquartile range for the whitening factor. The last row is the distance between two different tasks' metrics from the same draw, the scale the others are read on.

Two draws of $N{=}512$ examples	raw C	$\operatorname{cond}(C; r)$	C^p
per-task C_t (1152 pairs)	0.064	0.062	0.065 [0.051, 0.095]
pooled $\bar{C}$ (144 pairs)	0.028	0.028	0.031 [0.024, 0.044]
different tasks, same draw (1344 pairs)	1.139	1.112	0.547 [0.440, 0.636]

Pooling, not conditioning, averages the sample out. Conditioning leaves the redraw distance essentially unchanged ($0.064 \to 0.062$); its role is the bound on κ of Appendix E. Pooling over the $T = 8$ tasks reduces the distance by $2.3\times$ (0.028 on the raw metric, 0.031 on $\bar{C}^p$), close to the $\sqrt{8}$ that independent averaging would give. The pooled metric, which the orthogonalization step inverts and which Figure 2 identifies as the carrier of the gain, is thus the most stable object in the pipeline, $40\times$ below the cross-task distance it encodes, and its condition number changes by 0.35% in the median and at most 4.1% across draws.

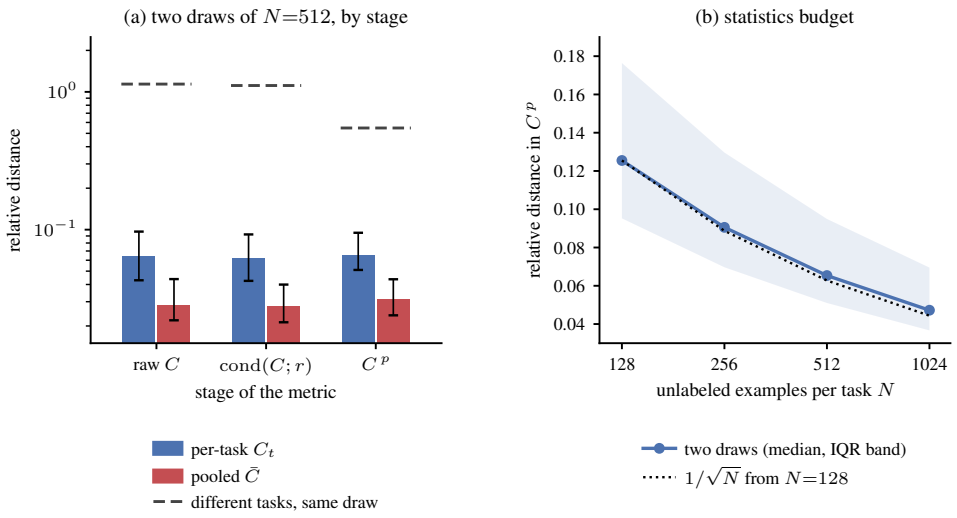

Figure 3: **Sample-level stability of the input metric (ViT-B/32**, T=8). (a) Median relative distance between two independent draws of N=512 examples at the three stages the merge puts C through, for the per-task and the pooled metric (bars with interquartile whiskers, log scale); the dashed marks are the distance between two different tasks' metrics from the same draw. (b) The same distance on C^p against the budget N, with its interquartile band and the $1/\sqrt{N}$ law anchored at N=128.

Other ways the sample can differ. Table 10 replaces the second random draw by other meanings of "a different sample". Class-balanced draws, the choice between random and balanced sampling, and the choice of split are indistinguishable from a plain redraw (0.063 to 0.074 on C^p). The one choice that moves the metric materially is whose activations define it: through the finetuned expert instead of the base encoder the distance is 0.378, close to the different-task level of 0.547. Every reported metric is collected through the shared base.

Table 10: **Other meanings of a different sample.** Median relative distance between two per-task estimates on the whitening factor C^p (**ViT-B/32**, T=8), for each way in which the two estimates can differ; the different-task distance is the reference.

What differs between the two estimates (per-task, on C^p)	distance
which 512 examples, random draws	0.065
which 512 examples, class-balanced draws	0.063
random vs. class-balanced draw	0.068
training vs. validation vs. test split	0.074
base encoder vs. finetuned expert	0.378
different task (reference)	0.547

The estimator follows $1/\sqrt{N}$. From $N = 128$ to 1024 the redraw distance on C^p falls as 0.125, 0.091, 0.065, 0.047, within a few percent of the $1/\sqrt{N}$ law anchored at $N = 128$ (0.125, 0.089, 0.063, 0.044), and the kept subspace converges monotonically, from 4.99× to 5.77× chance (Table 11, Figure 3b). Two disjoint halves of one permutation, an independence mechanism unrelated to reseeding, give 0.092 at $N = 256$ against 0.091 across seeds, so the seed-to-seed spread is sampling noise and nothing else. Doubling the budget to 1024 would reduce an already second-order quantity by 1.4×.

Table 11: **The statistics budget.** Distance between two draws on C^p against the budget N (**ViT-B/32**, T=8), the $1/\sqrt{N}$ law anchored at N=128, and the agreement of the kept top-r eigenspace as a multiple of chance. At N=256, two disjoint halves of one permutation give 0.092.

Budget N	128	256	512	1024
two draws, on C^p	0.125	0.091	0.065	0.047
$1/\sqrt{N}$ from N=128	0.125	0.089	0.063	0.044
top-r overlap ($\times$ chance)	4.99	5.36	5.60	5.77

Accuracy over draws. Table 12 measures how much the merged model itself moves: five independent draws (seeds 0 to 4) of the $N = 512$ examples per task, each merged and evaluated with its own validation-selected α, on **ViT-B/32** (T=8) and on GLUE. The standard deviation of the normalized accuracy of F-TSV is 0.11 on vision and 0.48 on GLUE, and the selected α varies by at most 0.2 across draws. The mean increment of the pooled orthogonalization over selection alone, 3.76 and 7.97 normalized points, exceeds the standard deviation of either row by more than an order of magnitude.

Table 12: **Accuracy over draws of the statistics sample.** Normalized test accuracy over five independent draws (seeds 0 to 4) of the N=512 examples per task, mean $\pm$ standard deviation, each draw merged and evaluated with its own validation-selected α, whose range is given below it. Δ_{sel} is metric-guided selection without the pooled orthogonalization, as in Table 16.

		Δ_{sel}	F-TSV	F-TSV^2
ViT-B/32, T=8	normalized, mean $\pm$ sd	92.09 ± 0.12	95.85 ± 0.11	95.97 ± 0.07
	selected α range	1.0–1.1	1.6–1.7	1.6–1.8
RoBERTa (GLUE)	normalized, mean $\pm$ sd	79.89 ± 0.17	87.86 ± 0.48	87.91 ± 0.31
	selected α range	1.4–1.5	3.0–3.2	3.2–3.4

The metric-level study covers one backbone, one task count and the input metric only; the accuracy spread is measured on **ViT-B/32** (T=8) and on GLUE.

N THE WHITENING POWER: FULL GRID

Table 13 extends the grid of §5 to every power that was run, with the coefficient each cell selected. All cells of a row share the cached statistics: the power is applied to the conditioned metric at merge time, not at collection time. On GLUE the cells at $p \notin \{0, \frac{1}{2}\}$ were swept on the coarser grid $\{0.5, 1.0, \ldots, 6.0\}$; the $p = \frac{1}{2}$ cell selects $\alpha = 3.0$ on both grids and is the run of Table 2. Choosing p by the validation objective instead of by derivation returns $\frac{1}{2}$ on every modality: the validation optimum over α is highest at $p = \frac{1}{2}$ on **ViT-B/32** and on GLUE, and the validation likelihood is lowest there on MergeBench. The collapse at large p is the orthogonalization's inverse factor at work: selection never multiplies kept content by a power of C, whereas equation 8 amplifies by up to $\kappa(\bar{C})^p$, which on **ViT-B/32**'s worst-conditioned layer is 38 at $p = \frac{1}{2}$ and $\sim 10^{12}$ at $p = 4$.

O SITE ATTRIBUTION: DEFINITIONS AND THE GLUE PANELS

Sites and groups. A site is a weight matrix whose input (for C) or output (for G) is observable by a forward hook on the base encoder. On the CLIP encoders these are, per block, the fused attention in-projection (q/k/v), the attention out-projection and the two MLP linears, 48 matrices on **ViT-B/32** (12 blocks); on **RoBERTa-base** the query, key, value and attention-output projections and the two feed-forward linears, 72 matrices; on **Llama-3.2-3B** the q, k, v and o projections and the gate, up and down projections, 196 matrices (28 layers). The groups of Tables 3 and 14 are: q/k/v, the attention

Table 13: **The whitening power, full grid.** Every cell is one run at its own validation-selected α (second line of each block), on the conditioning recipe of Appendix E. Vision and GLUE report absolute accuracy with normalized in parentheses; MergeBench the five-pillar average and the teacher-forced validation likelihood at the selected α. The p=0 column is the metric switched off: the c_{sites}=`none` control on vision and GLUE, and F-TSV run at p=0 on MergeBench, which Proposition 3 makes TSV-M and which is a separate evaluation from the TSV-M row of Table 4. †: the validation optimum sat on the grid floor (α=0.1), so the cell bounds the power rather than measuring it. **Bold** is the best cell per row.

p	0 (TSV-M)	0.25	0.5 (derived)	1.0 (PAVE)	2.0
ViT-B/32 T=8	83.41 (92.05)	86.47 (95.52)	**86.81 (96.02)**	85.92 (95.07)	84.93 (93.94)
selected α	0.9	1.5	1.6	1.6	1.3
RoBERTa GLUE	66.35 (78.01)	71.28 (85.29)	**73.13 (87.86)**	68.40 (82.52)	56.60 (68.16)
selected α	1.4	2.0	3.0	3.0	1.5
Llama-3.2-3B MergeBench	39.37	39.67	**39.76**	36.46	13.85†
selected α	0.8	1.3	1.8	1.7	0.1†
validation NLL	1.6869	1.6415	1.6404	1.6666	8.5171

Sites kept	**Input metric** C RoBERTa, GLUE		**Sites kept**	**Output metric** G RoBERTa, GLUE	
	Score	Gain		Score	Gain
none (≡ TSV-M)	$66.35_{(78.01)}$	—	none (≡ F-TSV)	$73.13_{(87.86)}$	—
q/k/v	$69.84_{(82.79)}$	+4.78	q/k/v	$72.54_{(87.41)}$	−0.45
all attention	$70.45_{(83.73)}$	+5.72	all attention	$72.80_{(87.70)}$	−0.16
non-residual	$71.41_{(84.54)}$	+6.53	non-residual	$72.99_{(87.84)}$	−0.02
residual	$68.20_{(80.94)}$	+2.93	residual	$73.39_{(88.30)}$	+0.44
q/k/v + residual	$70.65_{(85.14)}$	+7.13	q/k/v + residual	$73.05_{(87.87)}$	+0.01
MLP linears	$69.72_{(82.58)}$	+4.57	MLP linears	$73.96_{(88.95)}$	+1.09
MLP + residual	$69.99_{(83.35)}$	+5.33	MLP + residual	$73.83_{(88.89)}$	+1.03
all sites	$\mathbf{73.13_{(87.86)}}$	+9.85	all sites	$\mathbf{73.89_{(88.98)}}$	+1.12

Table 14: **Site attribution on GLUE: the same ablation as Table 3, on the other modality.** RoBERTa-base, 8 GLUE tasks, both panels; which weight matrices keep their measured metric, with dropped sites behaving exactly as the method one rung down (Proposition 3). Absolute score with normalized in subscript; *Gain* is the normalized-score change against none. The output panel uses the Gaussian probes on the encoder's final representation, as every two-sided result in this paper (Appendix Q).

projections that do not write into the residual stream; all attention, those plus the attention out-projection; MLP linears, every non-attention site; residual, the two matrices per block whose output is added to the residual stream (attention out-projection and MLP down-projection); non-residual, its complement; and unions of these. A site outside the kept group receives the identity metric on that side and, by Proposition 3, is merged there exactly as by the method one rung down. Matrices that are not sites (token and positional embeddings, the convolutional stem, projection heads) take the Euclidean path; on **Llama-3.2-3B** the token embedding is the exception treated in Appendix L. Table 14 is the GLUE counterpart of Table 3, discussed in §5.

P THE WHITENING POWER EXTENDED RESULTS

To clarify the role of the whitening power, we provide additional results here to clarify §5 in the main manuscript. The power enters twice, and the derivation constrains only one of the two: selection uses C_t^p to rank directions, which any $p > 0$ does, whereas the orthogonalization's constraint $R\bar{C}^{2p}R^\top = I$ coincides with $\bar{C}$-orthonormality only at $p = \frac{1}{2}$ (Proposition 2). Sweeping the two independently over $\{0, \frac{1}{4}, \frac{1}{2}, 1, 2\}^2$ separates them: on both **ViT-B/32** (T=8) and GLUE the orthogonalization power $\frac{1}{2}$ is optimal in *every* row of the grid, for every selection power, while the selection power is comparatively flat (96.13 at p_{sel}=$\frac{1}{4}$ against 96.02 at $\frac{1}{2}$ on vision, inside the 0.11 standard deviation of the draw). The quantitative prediction the derivation makes is therefore confirmed exactly where it is made, and the earlier single-p sweep was measuring the orthogonalization.

Refining the grid to $\{0.3, 0.4, 0.5, 0.6, 0.7\}$ places the vision optimum at p=0.4 (96.07) against 96.02 at $\frac{1}{2}$, a separation of 0.05 points not distinguishable inside the 0.11 standard deviation of the statistics draw. On GLUE the optimum is $\frac{1}{2}$ (87.86, against 87.68 at 0.4). We therefore claim that the derived exponent is not improved upon, rather than that it is uniquely optimal.

Q EXPERIMENTAL PROTOCOL AND REPRODUCTION

Backbones, experts and task suites. *Vision:* OpenCLIP ViT-B/32, ViT-B/16 and ViT-L/14 image encoders with the public per-task experts used by MASS (Crisostomi et al., 2026). The 8-task suite (Ilharco et al., 2023) is SUN397 (Xiao et al., 2016), Cars (Krause et al., 2013), RESISC45 (Cheng et al., 2017), EuroSAT (Helber et al., 2019), SVHN (Netzer et al., 2011), GTSRB (Stallkamp et al., 2011), MNIST (LeCun et al., 1998) and DTD (Cimpoi et al., 2014); the 14-task suite (Wang et al., 2024) adds Flowers102 (Nilsback & Zisserman, 2008), PCAM (Veeling et al., 2018), FER2013 (Goodfellow et al., 2013), OxfordIIITPet (Parkhi et al., 2012), STL10 (Coates et al., 2011) and CIFAR100 (Krizhevsky & Hinton, 2009); the 20-task suite adds CIFAR10 (Krizhevsky & Hinton, 2009), Food101 (Bossard et al., 2014), FashionMNIST (Xiao et al., 2017), EMNIST (Cohen et al., 2017), KMNIST (Clanuwat et al., 2018) and RenderedSST2 (Socher et al., 2013). Each task is evaluated with its frozen zero-shot CLIP head. *Language:* `roberta-base` with the RoBERTa-base GLUE experts distributed with WUDI-Merging (Cheng et al., 2025) and reused by ESM (Li et al., 2026), on CoLA, SST-2, MRPC, STS-B, QQP, MNLI, QNLI and RTE (Wang et al., 2018); each expert's classification head is split from its checkpoint and swapped in at evaluation, never merged. Every baseline row of Tables 1 and 2 was produced by the same evaluation code and from the same experts downloaded from Huggingface, under the coefficient protocol above, and each cell is a single run whose configuration is stored beside its score. The vision experts are the publicly accessible per-task checkpoints on Huggingface released by Crisostomi et al. (2026). Because the normalized score is divided by the same reference for every method, that offset cancels in the comparison that the tables make. *Generative:* `meta-llama/Llama-3.2-3B` in bf16 with the five MergeBench experts `Llama-3.2-3B-`{`instruction, math, coding, multilingual, safety`} (He et al., 2026), the embedding and output are the same one matrix and are merged once.

Splits and scores. On vision and GLUE the validation split is a seeded 10% of each task's training set, the test split is the standard test set (for GLUE, the original validation set, MNLI uses the matched set), and the statistics are drawn from the remaining training data. Scores are accuracy, except Matthews correlation for CoLA and Spearman correlation for STS-B, floored at 0 as in the GLUE tables. The normalized score divides each task's merged score by its expert's score and averages over tasks. On MergeBench, the statistics and the coefficient sweep both use the benchmark's per-domain held-out sets (1000 rows per domain under the alpaca SFT template), and the five pillars are scored on the benchmark's test tasks as described.

Statistics budgets and defaults are in Table 15. All statistics are collected through the base model, in one forward pass per batch for C and one forward plus n_p backward passes per batch for G, over token rows with padding masked; the Gaussian probes are drawn on the encoder's final representation (the CLIP image embedding; the final hidden states of the RoBERTa body and of the Llama decoder). The vision and GLUE rank budgets are absolute; the **Llama-3.2-3B** ones are fractions of each layer's width (Appendix E). The rank kept per task and layer is $k = \max(1, \lfloor \min(m, n)/T \rfloor)$, so the T stacked blocks fill the layer's spectrum exactly once.

Table 15: Statistics budgets and defaults of every reported cell. Examples are counted per task; a batch of G carries its n_p probes. Ranks on the encoders are absolute; on **Llama-3.2-3B** they are fractions of each layer's width, resolved per layer.

		Vision (CLIP), GLUE (RoBERTa)	MergeBench (**Llama-3.2-3B**)
input metric C	examples per task	$4 \times 128 = 512$	$63 \times 16 = 1008$ sequences (≤ 4096 tokens)
	rank r	128	$0.2\times$ width (614 of 3072; 1638 of 8192)
output metric G	examples per task	$8 \times 64 = 512$	$128 \times 2 = 256$ sequences
	probes per batch n_p	32	8
	rank r_g	256	$0.4\times$ width
token-embedding metric		—	128×2 sequences, 8 probes
whitening powers	$p = p_g$	$\frac{1}{2}$	$\frac{1}{2}$
activations		base encoder, training split	base model, held-out sets

Table 16: Selected coefficients α of the reported cells (Tables 1, 2, and 4 and Figure 2). TSV-M is the c_{sites}=`none` control and Δ_{sel} the metric-guided selection without the pooled orthogonalization. Grids: 0.1 to 2.5 on vision, to 4.0 on GLUE, to 2.0 on MergeBench.

	ViT-B/32			**ViT-B/16**			**ViT-L/14**			GLUE	MergeBench
T	8	14	20	8	14	20	8	14	20	8	5
TSV-M	0.9	0.9	0.8	0.9	0.9	0.8	1.1	1.0	1.0	1.4	0.8
Δ_{sel}	1.1	1.0	0.9	1.0	1.0	0.9	1.3	1.1	1.1	1.4	0.8
F-TSV	1.6	1.6	1.5	1.5	1.7	1.6	2.2	2.1	1.8	3.0	1.8
F-TSV^2	1.6	1.8	1.6	1.7	1.7	1.5	1.9	1.9	1.9	3.2	1.8

Coefficient selection. The candidate grid is a property of the benchmark: $\{0.1, 0.2, \ldots, 2.5\}$ on vision, $\{0.1, \ldots, 4.0\}$ on GLUE and $\{0.1, \ldots, 2.0\}$ on MergeBench, never including 0, which is the pretrained model rather than a merge. The vision ceiling is set by **ViT-L/14**, whose optima reach 2.2, and the GLUE ceiling by Iso-CTS and F-TSV^2, whose optima exceed 3.0. For vision and GLUE, each candidate is evaluated on the validation split, and the one with the highest average normalized score is retained. On MergeBench, a per-α generative evaluation is too expensive, so each candidate is scored by the mean teacher-forced negative log-likelihood of the held-out rows (prompt and response tokens, up to 1024 tokens, equal-weighted over the five domains), and the lowest-likelihood α receives the single full evaluation. Every results file flags an optimum that sits on a grid boundary. One of the results reported here is flagged, at the grid's smallest coefficient $\alpha = 0.1$: the p=2.0 MergeBench cells of Table 13. Table 16 lists the selected coefficient of every cell of Tables 1, 2, and 4 produced by our methods and their in-family controls, none of which sit on a boundary.

Observability. Statistics come from forward hooks on the base encoder. On the CLIP encoders the fused attention module is temporarily replaced by an unfused equivalent at collection time, so that both attention projections are observable on both sides. Matrices without observed statistics take the Euclidean path per Proposition 3; non-matrix parameters (biases, normalization gains) are averaged across tasks.

Determinism, and the MergeBench spread. Given fixed experts, cached statistics and the fixed validation split the merge is deterministic: three runs at different global return identical scores to every stored digit, and the vision and GLUE evaluations, being deterministic passes conducted on fixed splits, they inherit that determinism, so we quote no confidence intervals there. The MergeBench evaluation does not: two of its five pillars are scored by sampled generation. One configuration (F-TSV at $p = \frac{1}{2}$ with the embedding metric, $\alpha = 1.8$) evaluated three times returned five-pillar averages of 39.67, 39.76 and 39.80: coding and multilingual were identical across the three, instruction and math moved by 0.21 and 0.15, and safety by 0.68. We treat differences below about 0.2 five-pillar points as not separating two MergeBench configurations. This covers the gap between

F-TSV and F-TSV^2 in Table 4, and the gap between that table's F-TSV row and the $p = \frac{1}{2}$ cell of Table 13.

MergeBench evaluation. *Instruction:* IFEval, the mean of prompt-level and instruction-level strict accuracy. *Math:* GSM8K with 8-shot chain of thought, flexible-extract exact match. *Multilingual:* the twelve Okapi tasks (M-MMLU, M-ARC and M-HellaSwag in French, Spanish, German and Russian), accuracy, zero-shot. These three run in lm-eval-harness with a 4096-token context and an automatically sized batch. *Coding:* HumanEval+ and MBPP+ through the bigcode evaluation harness, pass@1 over 10 samples at temperature 0.2, top-p 0.95, 512 new tokens, bf16, averaged over the two suites. *Safety:* WildGuardTest (harm rate), HarmBench and DoAnythingNow (attack success rate) and XSTest (accuracy on its safe/unsafe contrast set) through safety-eval-fork with the Llama-3 input template; the first three are inverted so that higher is safer and the four are averaged. Base model, task list, few-shot counts, sampling configuration and safety flags are those of MergeBench's own evaluation script, and every run persists every metric key lm-eval reports, so alternative readings (normalized accuracy on M-ARC and M-HellaSwag, prompt-level-only IFEval) can be produced without a re-run.

Cost. Everything is training-free. The one-time per-task cost is one forward pass over the statistics sample for C and, two-sided, one forward plus n_p backward passes per batch for G; statistics are cached and shared by every configuration, so an ablation cell costs one merge plus one evaluation. Relative to TSV-M, merging adds per layer one eigendecomposition per active metric and two thin factorizations (a QR and a second SVD of the kept block). On one RTX PRO 6000 Blackwell, collecting C for the eight vision tasks takes about three minutes; on **Llama-3.2-3B** the token-embedding matrix is merged in 5.2 seconds using 15.3 GiB (Appendix L). The merged model is a single static checkpoint in the base architecture, with no routing, no extra parameters and no inference-time overhead.