Merging Adapted Models via Data-Free Covariance Estimation

Anonymous Authors[1]

Abstract

Model merging provides a way of cheaply combining individual models to produce a model that inherits each individual's capabilities. While some merging methods can approach the performance of multitask training, they are often heuristically motivated and lack theoretical justification. A principled alternative is to pose model merging as a layer-wise optimization problem that directly minimizes interference between tasks. However, this formulation requires estimating per-layer covariance matrices from data, which may not be available when performing merging. In contrast, many of the heuristically-motivated methods do not require auxiliary data, making them practically advantageous. In this work, we revisit the interference minimization framework and show that, under certain conditions, covariance matrices can be estimated directly from *difference matrices*, eliminating the need for data while also reducing computational costs. We validate our approach across vision and language benchmarks on models ranging from $86M$ parameters to $7B$ parameters, outperforming previous data-free state-of-the-art merging methods.

1. Introduction

Large-scale pretrained models have become the backbone of modern machine learning (Bommasani et al., 2021), and fine-tuning them for specific downstream tasks is now standard practice. This has led to a proliferation of publicly available task-specific expert models (Wolf et al., 2020), each excelling in a narrow domain. However, many downstream applications demand capabilities that span multiple domains. Multitask learning (Caruana, 1997) and model ensembling are natural candidates for combining such capabilities, but the former requires simultaneous access to all training datasets, while the latter incurs significant storage and inference overhead at deployment. In contrast, model merging combines expert capabilities by directly merging their parameters (Regent's et al., 1996; Matena & Raffel, 2022; Wortsman et al., 2022).

Despite their empirical success, state-of-the-art merging methods, such as TIES (Yadav et al., 2023), Iso-C (Marczak et al., 2025) and TSV (Gargiulo et al., 2025) remain largely based off heuristics and lack theoretical guarantees. A notable exception is RegMean (Jin et al., 2023), which frames model merging as a tractable layer-wise optimization objective. However, RegMean involves computing covariance matrices for each layer across all the tasks being merged, which requires access to each task's data when performing merging. For most publicly available expert models, this training data is not released. Even when data is available, computing and storing these matrices becomes prohibitively expensive for large-scale models. This limits RegMean's applicability in precisely the settings where model merging is most attractive.

This raises a natural question, *can the covariance matrices required by RegMean be estimated without access to data?* In this work, we answer in the affirmative: under certain conditions, covariance matrices can be recovered directly from each task's *difference matrix* (i.e., the difference between fine-tuned and pretrained matrices). We call this estimator "**A**pproximating **C**ovariances via **T**ask Vectors for Activation **Mat**ching" (ACTMat). Combining ACTMat with the RegMean objective, we obtain a fully data-free merging method that consistently outperforms prior state-of-the-art data-free approaches, as illustrated in Figure 1.

2. Related Work

A number of merging methods are based on the principle of *interference minimization*. TIES (Yadav et al., 2023) reduces interference at the parameter level by trimming low-magnitude parameters and resolving sign conflicts across task vectors. Similarly, DARE (Yu et al., 2024) resets a fraction of fine-tuned parameters to their original weights at random. Another family of methods leverages the matrix structure of linear layers via the Singular Value Decomposition (SVD). Task Singular Vectors (TSV) (Gargiulo et al., 2025) reduces "Singular Task Interference" by decor-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the ICML 2026 Workshop "Continual Adaptation at Scale: Towards Sustainable AI". Do not distribute.

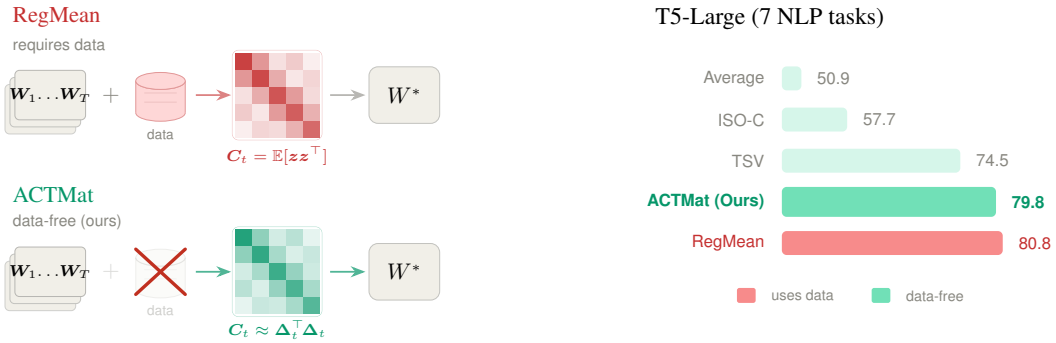


Figure 1. **Top:** RegMean requires data to compute activation covariances $\boldsymbol{C}_t$, while ACTMat estimates them directly from the difference matrices as $\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t \approx \mathbf{C}_t$. **Bottom:** On T5-Large, ACTMat nearly matches RegMean's accuracy without any data, while substantially outperforming other data-free baselines.

relating singular vectors of different tasks before merging, while Iso-C (Marczak et al., 2025) flattens the spectrum of the merged matrix. KnOTS (Stoica et al., 2025) merges LoRA-fine-tuned models in an aligned space via the SVD. Methods such as TSV and Iso-C are performant in the absence of data, while others such as RegMean (Jin et al., 2023), LOT (Sun et al., 2025), WUDI (Cheng et al., 2025), and AdaMerging (Yang et al., 2024) require auxiliary data either for optimization or hyper-parameter tuning. Similarly to TSV and Iso-C, ACTMat is entirely data-free.

3. Method

Background and Notation. We consider the standard model merging setting in which T models with the same architecture and pretrained initialization are fine-tuned on different tasks, then combined into a single model. Specifically, let $f : \mathcal{X} \times \Theta \to \mathcal{Y}$ be a neural network parameterized by $\boldsymbol{\theta} \in \Theta$. Each task t is associated with a discrete distribution $\mathcal{D}_t$ over $\mathcal{X}$. Fine-tuning on task t involves updating the model parameters, starting at initial parameters $\boldsymbol{\theta}_0$ and ending with task-specific parameters $\boldsymbol{\theta}_t$. For an arbitrary linear layer in f, we denote its pretrained parameters by $\boldsymbol{W}_0 \in \mathbb{R}^{D_o \times D_i}$, its fine-tuned parameters for task t by $\boldsymbol{W}_t \in \mathbb{R}^{D_o \times D_i}$, its input by $\boldsymbol{z} \in \mathbb{R}^{D_i}$, its output by $\boldsymbol{y} = \boldsymbol{W}\boldsymbol{z} \in \mathbb{R}^{D_o}$, and its difference matrix by $\boldsymbol{\Delta}_t := \boldsymbol{W}_t - \boldsymbol{W}_0$. We use $\boldsymbol{x}$ to denote the inputs to the model and $\boldsymbol{x} \sim \mathcal{D}_t$ to indicate that they are sampled according to the t-th distribution. With some abuse of notation we also use $\boldsymbol{z} \sim \mathcal{D}_t$ to indicate the induced distribution over an arbitrary linear layer's inputs, when model inputs $\boldsymbol{x}$ are sampled from the t-th distribution.

3.1. Model Merging as Interference Minimization

Following Jin et al. (2023), we formulate model merging as a layer-wise optimization problem. For each linear layer, we seek the merged weights that best preserve each task's activations:

$$\boldsymbol{W}^\star \in \arg\min_{\boldsymbol{W}} \sum_{t=1}^{T} \mathbb{E}_{\boldsymbol{z}\sim\mathcal{D}_t}\left[\|\boldsymbol{W}\boldsymbol{z} - \boldsymbol{W}_t\boldsymbol{z}\|_2^2\right] \quad (1)$$

This objective is solved independently per layer and admits

$$\boldsymbol{W}^\star = \sum_{t=1}^{T} \boldsymbol{W}_t \boldsymbol{C}_t \Big(\sum_{t'} \boldsymbol{C}_{t'}\Big)^\dagger, \quad (2)$$

as the minimum Frobenius norm solution (proof in Appendix), where $\boldsymbol{C}_t = \mathbb{E}_{\boldsymbol{z}\sim\mathcal{D}_t}[\boldsymbol{z}\boldsymbol{z}^\top]$ denotes the second moment of the layer inputs under distribution $\mathcal{D}_t$ and † the Moore–Penrose pseudoinverse. Under this formulation, model merging reduces to a covariance estimation problem.[1] In the following section, we show that, under certain conditions, the covariance matrix can be approximated from the difference matrices as $\boldsymbol{C}_t \approx \boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t$, yielding a **fully data-free merging rule**:

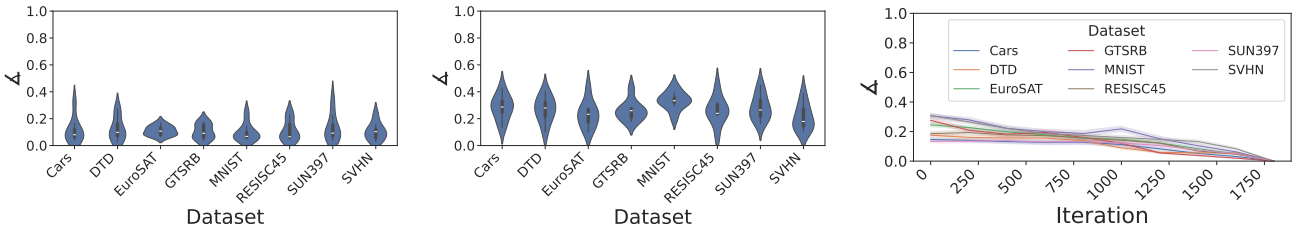
$$\boxed{\boldsymbol{W}^\star \approx \sum_{t=1}^{T} \boldsymbol{W}_t \Big(\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t\Big)\Big(\sum_{t'=1}^{T} \boldsymbol{\Delta}_{t'}^\top \boldsymbol{\Delta}_{t'}\Big)^\dagger.} \quad (3)$$

3.2. Covariance Estimation of Activations

In this section, we show that the covariance matrices of activations in Equation (2) can be approximated directly from the difference matrices, up to a scaling factor. In other words, we show that the *angular distance* between $\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t$ and $\boldsymbol{C}_t$ is small, where $\measuredangle(\boldsymbol{A}, \boldsymbol{B}) := \arccos(\langle \boldsymbol{A}, \boldsymbol{B}\rangle_F / (\|\boldsymbol{A}\|_F \|\boldsymbol{B}\|_F))$ denotes the angular distance metric.[2] Consider a linear layer fine-tuned using full-batch gradient descent for K iterations with a fixed learning rate η. Let $\boldsymbol{z}^{(k)}$ denote the layer's input at iteration k, $\boldsymbol{y}^{(k)} = \boldsymbol{W}^{(k)}\boldsymbol{z}^{(k)}$ its output, and $\boldsymbol{g}^{(k)} := \nabla_{\boldsymbol{y}} \mathcal{L}(\boldsymbol{x}; \boldsymbol{\theta}^{(k)})$

[1] Strictly speaking, $\boldsymbol{C}_t$ is a second moment matrix rather than a centered covariance, though we refer to $\boldsymbol{C}_t$ as a covariance matrix throughout for brevity.

[2] $\langle \boldsymbol{A}, \boldsymbol{B}\rangle_F := \mathrm{tr}(\boldsymbol{A}^\top \boldsymbol{B})$ denotes the Frobenius inner product and $\|\boldsymbol{A}\|_F := \sqrt{\langle \boldsymbol{A}, \boldsymbol{A}\rangle_F}$ the associated norm.



(a) Cross-term error *(b)* Correlation error *(c)* Drift error

Figure 2. Empirical measurement of the three angular error terms in Theorem 3.1 on ViT-B/16. **(a)** Cross-term error $\epsilon^{(\text{cross})}$. **(b)** Correlation error $\epsilon^{(\text{corr})}$. **(c)** Drift error $\epsilon^{(\text{drift})}$ measured during training. All three terms remain small across layers and tasks, indicating that $\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t$ is well-aligned with the final covariance $\boldsymbol{C}_t^{(K)}$.

the gradient of the loss with respect to the output. Using the chain rule, the gradient with respect to $\boldsymbol{W}$ at iteration k is $\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_t}\left[\nabla_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{x};\boldsymbol{\theta}^{(k)})\right] = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_t}\left[\boldsymbol{g}^{(k)}\boldsymbol{z}^{(k)\top}\right]$. Since $\boldsymbol{\Delta}_t = -\eta\sum_{k=0}^{K}\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_t}[\boldsymbol{g}^{(k)}\boldsymbol{z}^{(k)\top}]$, one can check that

$$\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t \propto \sum_{k,k'}\mathbb{E}_{\boldsymbol{x},\boldsymbol{x}'\sim\mathcal{D}_t}\left[\boldsymbol{z}^{(k)}\boldsymbol{z}'^{(k')\top}\boldsymbol{g}^{(k)\top}\boldsymbol{g}'^{(k')}\right], \quad (4)$$

suggesting that the product $\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t$ *captures* second-order statistics of the layer's inputs. However, recovering the covariance of activations at the end of training, $\boldsymbol{C}_t^{(K)} = \mathbb{E}[\boldsymbol{z}^{(K)}\boldsymbol{z}^{(K)\top}]$, from this expression is not immediate. The following theorem shows that the angular distance between $\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t$ and $\boldsymbol{C}_t^{(K)}$ is upper bounded by three error terms (proof in Appendix). In practice, we find that each of these three error terms is relatively small, indicating that $\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t$ is approximately proportional to $\boldsymbol{C}_t^{(K)}$.

Theorem 3.1 (Covariance Estimation)**.** *Consider a linear layer fine-tuned using gradient descent for K iterations with learning rate η, and let $\boldsymbol{z}^{(k)}$, $\boldsymbol{g}^{(k)}$ denote the layer's input and its output gradient at iteration k, respectively. Define*

$$\overline{\boldsymbol{G}} := \sum_{k=0}^{K}\mathbb{E}\left[\boldsymbol{g}^{(k)}\boldsymbol{z}^{(k)\top}\right],$$
$$\overline{\boldsymbol{S}} := \sum_{k=0}^{K}\mathbb{E}\left[\boldsymbol{z}^{(k)}\boldsymbol{z}^{(k)\top}\|\boldsymbol{g}^{(k)}\|^2\right],$$
$$\widetilde{\boldsymbol{S}} := \sum_{k=0}^{K}\mathbb{E}\left[\boldsymbol{z}^{(k)}\boldsymbol{z}^{(k)\top}\right]\mathbb{E}\left[\|\boldsymbol{g}^{(k)}\|^2\right],$$

where the expectation is taken over $\mathcal{D}_t$, and $\|\cdot\|$ denotes the Euclidean norm. Then, the angular distance between $\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t$ and the final covariance $\boldsymbol{C}_t^{(K)}$ satisfies

$$\measuredangle\left(\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t,\ \boldsymbol{C}_t^{(K)}\right) \le \epsilon^{(\text{cross})} + \epsilon^{(\text{corr})} + \epsilon^{(\text{drift})}, \quad (5)$$

where $\epsilon^{(\text{cross})} = \measuredangle\left(\overline{\boldsymbol{G}}^\top\overline{\boldsymbol{G}},\ \overline{\boldsymbol{S}}\right)$ is the cross-term error, $\epsilon^{(\text{corr})} = \measuredangle\left(\overline{\boldsymbol{S}},\ \widetilde{\boldsymbol{S}}\right)$ is the correlation error, and $\epsilon^{(\text{drift})} = \measuredangle\left(\widetilde{\boldsymbol{S}},\ \boldsymbol{C}_t^{(K)}\right)$ is the drift error. In particular, $\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t \propto \boldsymbol{C}_t^{(K)}$ when all three errors vanish.

To analyze the contributions of each of the error terms in Theorem 3.1, we fine-tune the ViT-B/16 (Dosovitskiy et al., 2021) model on eight downstream tasks, and analyze the error terms in Theorem 3.1. (See Appendix A for details).

The *cross-term error* $\epsilon^{(\text{cross})}$ arises due to off-diagonal contributions from the double summation over iterations and double expectation over samples in Equation (4). In Figure 2a, we report the angular distance between $\overline{\boldsymbol{G}}_t^\top\overline{\boldsymbol{G}}_t$ and $\overline{\boldsymbol{S}}$ across all datasets and transformer layers for ViT-B/16, consistently finding negligible cross-term contributions.

The *correlation error* $\epsilon^{(\text{corr})}$ captures the coupling between per-sample activation outer products and output gradient norms, and this error term vanishes when these quantities are uncorrelated. In Figure 2b, we consistently find relatively small angular distances between $\overline{\boldsymbol{S}}$ and $\widetilde{\boldsymbol{S}}$ across datasets and transformer layers. Interestingly, a similar error term is encountered in KFAC (Martens & Grosse, 2015), where activations and output gradients are assumed to be uncorrelated. In contrast, the correlation error in Theorem 3.1 vanishes as the correlation between activations and *output gradient norms* approaches zero.

The *drift error* $\epsilon^{(\text{drift})}$ reflects how much the activation covariances change over the course of training, and is small when the covariances remain approximately stationary. In Figure 2c, we report the trajectory of angular distances between intermediate covariances $\boldsymbol{C}_t^{(k)}$ and the final covariance $\boldsymbol{C}_t^{(K)}$, observing low values and thus approximate stationarity. Altogether, these results suggest that $\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t$ is approximately proportional to the final covariance $\boldsymbol{C}_t^{(K)}$ in accordance with Theorem 3.1.

4. Experiments

We evaluate the performance of the ACTMat merge rule on vision and language tasks and on reasoning tasks.

Datasets & Models. For vision tasks, we follow Ilharco et al. (2021) and fine-tune ViT-B/16, ViT-B/32, and ViT-L/14 models on eight image classification datasets. For language tasks, we fine-tune T5-Base and T5-Large (Raffel et al., 2020) on seven multiple-choice datasets. For reasoning tasks, we evaluate merging models trained via reinforcement learning with verifiable rewards, using OLMo-

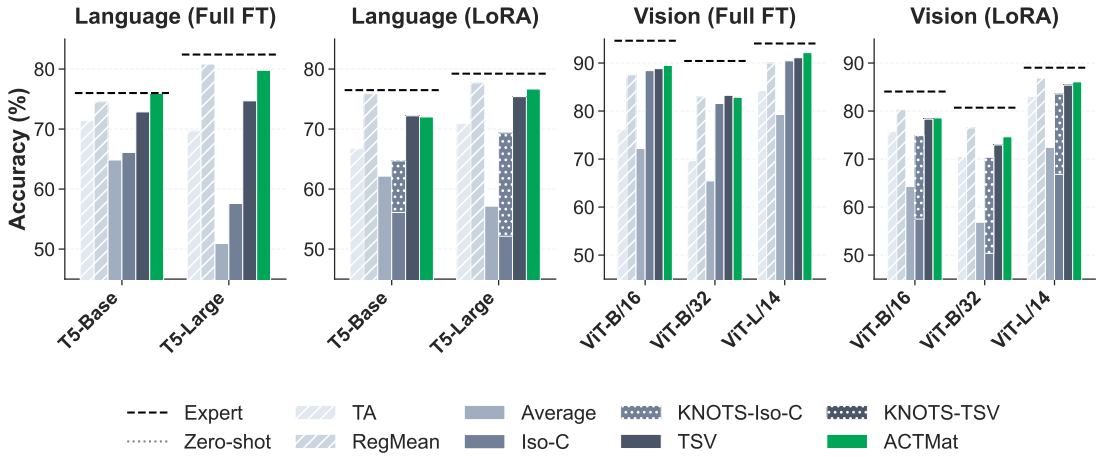


Figure 3. Comparison between test accuracy of merging methods across multiple settings (NLP models fine-tuned on 7 tasks and vision models fine-tuned on 8 tasks). *Hatched bars* indicate that the method is not data-free. *Stacked bars* with a dotted pattern indicate the performance of the method (bottom bar) and performance of the method when combined with KnOTS (Stoica et al., 2025) for LoRA fine-tuned models (improvement only for Iso-C).

Table 1. Comparison between merging methods in the RL Zero setting, where models are fine-tuned using RLVR starting from the OLMo-3-7B base model. All numbers are pass@1. We **bold** the best and underline the second best data-free method for each dataset (**HE**: HumanEval, **A24**: AIME 2024, **IF**: IFEval).

Method (↓)	HE	HE+	A24	A25	IF	Avg
ZERO-SHOT	50.8	47.3	22.1	21.6	30.7	34.5
EXPERT	62.0	57.0	38.2	30.7	82.4	54.1
REGMEAN	55.3	52.0	30.8	27.8	62.8	45.7
TA	57.2	52.1	36.8	31.1	32.2	41.9
AVERAGE	56.8	52.8	35.9	**32.3**	31.2	41.8
ISO-C	55.9	50.7	33.4	31.5	28.8	40.1
TSV	**59.0**	53.3	39.8	32.0	34.0	43.6
ACTMAT	58.2	**54.5**	**39.9**	29.8	**47.0**	**45.9**

3-7B (Olmo et al., 2025) as the base model. Specifically, we merge three publicly available RL-Zero checkpoints trained on math, code, and instruction-following. Following the evaluation protocol in Olmo et al. (2025), we evaluate mathematical reasoning on AIME 2024 & 2025. Coding and instruction-following abilities are evaluated using HumanEval (Chen et al., 2021), HumanEval+ (Liu et al., 2023) and IFEval (Zhou et al., 2023) (See Appendix A for details).

Baselines. We compare against several merging methods including simple weight averaging, Task Arithmetic (Ilharco et al., 2023), RegMean (Jin et al., 2023), Iso-C (Marczak et al., 2025), TSV (Gargiulo et al., 2025), and KnOTS (Stoica et al., 2025). We select TSV and Iso-C as they provide strong baselines in entirely data-free settings. Following Yadav et al. (2023), we also report the individual *Expert* performances, as well as the performance of the pretrained model which is referred to as the *Zero-shot* performance.

4.1. Results

Figure 3 compares ACTMat against baselines on vision and language tasks (full results in Appendix C). In the full fine-tuning setting, ACTMat achieves the highest average accuracy among all data-free methods on five out of six model configurations, outperforming the next best data-free method, TSV, by **+3.1** points on T5-Base and **+5.3** points on T5-Large. A similar trend holds under LoRA fine-tuning. Beyond supervised fine-tuning, Table 1 shows that ACTMat also outperforms data-free baselines on average when merging three OLMo-3-7B experts trained via RLVR (Olmo et al., 2025) on math, coding, and instruction following.

5. Conclusion

In this work, we presented ACTMat, a principled approach to data-free model merging that combines covariance estimates derived from difference matrices with the interference minimization framework of RegMean. Our approach leverages three empirical findings that make covariance approximation from difference matrices possible. Developing a theoretical understanding of why these properties hold is an interesting direction for future work, as is investigating applications of cheap, data-free covariance estimates beyond the merging setting.

References

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. On the opportunities and risks of foundation models, 2021.

Caruana, R. Multitask learning. *Mach. Learn.*, 28(1):41–75,

1997.

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. D. O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Cheng, G., Han, J., and Lu, X. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the Institute of Electrical and Electronics Engineers (IEEE)*, 2017.

Cheng, R., Xiong, F., Wei, Y., Zhu, W., and Yuan, C. Whoever started the interference should end it: Guiding data-free model merging via task vectors. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. Describing textures in the wild. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Cohen, D., Yang, L., and Croft, W. B. Wikipassageqa: A benchmark collection for research on non-factoid answer passage retrieval. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pp. 1165–1168, 2018.

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

Gargiulo, A. A., Crisostomi, D., Bucarelli, M. S., Scardapane, S., Silvestri, F., and Rodolà, E. Task singular vectors: Reducing task interference in model merging. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18695–18705, 2025. doi: 10.1109/CVPR52734.2025.01742.

Helber, P., Bischke, B., Dengel, A., and Borth, D. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2019.

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

Ilharco, G., Wortsman, M., Wightman, R., Gordon, C., Carlini, N., Taori, R., Dave, A., Shankar, V., Namkoong, H., Miller, J., Hajishirzi, H., Farhadi, A., and Schmidt, L. Openclip, July 2021. If you use this software, please cite it as below.

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

Jin, X., Ren, X., Preotiuc-Pietro, D., and Cheng, P. Dataless knowledge fusion by merging weights of language models. In *The Eleventh International Conference on Learning Representations*, 2023.

Khot, T., Clark, P., Guerquin, M., Jansen, P., and Sabharwal, A. Qasc: A dataset for question answering via sentence composition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8082–8090, 2020.

Krause, J., Stark, M., Deng, J., and Fei-Fei, L. 3d object representations for fine-grained categorization. In *International Conference on Computer Vision Workshops (ICML)*, 2013.

LeCun, Y. The mnist database of handwritten digits, 1998.

Levesque, H., Davis, E., and Morgenstern, L. The winograd schema challenge. *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*, 2012.

Liu, J., Xia, C. S., Wang, Y., and Zhang, L. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. *Advances in neural information processing systems*, 36:21558–21572, 2023.

Marczak, D., Magistri, S., Cygert, S., Twardowski, B., Bagdanov, A., and van de Weijer, J. No task left behind: Isotropic model merging with common and task-specific subspaces. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

Martens, J. and Grosse, R. Optimizing neural networks with kronecker-factored approximate curvature. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *JMLR: W&CP*, pp. 2408–2417, Lille, France, 2015.

Matena, M. and Raffel, C. Merging models with fisher-weighted averaging. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *Advances in Neural Information Processing Systems (NeurIPS) Workshops*, 2011.

Olmo, T., Ettinger, A., Bertsch, A., Kuehl, B., Graham, D., Heineman, D., Groeneveld, D., Brahman, F., Timbers, F., Ivison, H., et al. Olmo 3. *arXiv preprint arXiv:2512.13961*, 2025.

Raffel, C., Shazeer, N. M., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *ArXiv*, abs/1910.10683, 2020.

Regent's, ParkLondon, Ukj, and Utans, . Weight averaging for neural networksand local resampling. 1996.

Sakaguchi, K., Le Bras, R., Bhagavatula, C., and Choi, Y. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

Sharma, R., Allen, J., Bakhshandeh, O., and Mostafazadeh, N. Tackling the story ending biases in the story cloze test. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 752–757, 2018.

Stallkamp, J., Schlipsing, M., Salmen, J., and Igel, C. The german traffic sign recognition benchmark: a multi-class classification competition. In *International Joint Conference on Neural Networks (IJCNN)*, 2011.

Stoica, G., Ramesh, P., Ecsedi, B., Choshen, L., and Hoffman, J. Model merging with svd to tie the knots. *ICLR*, 2025.

Sun, W., Li, Q., Wang, W., Liu, Y., Geng, Y.-a., and Li, B. Towards minimizing feature drift in model merging: Layer-wise task vector fusion for adaptive knowledge integration. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Tafjord, O., Gardner, M., Lin, K., and Clark, P. Quartz: An open-domain dataset of qualitative relationship questions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 5941–5946, 2019.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics.

Wortsman, M., Ilharco, G., Li, M., Kim, J. W., Hajishirzi, H., and Farhadi, A. Robust fine-tuning of zero-shot models. In *CVPR*, 2022.

Xiao, J., Ehinger, K. A., Hays, J., Torralba, A., and Oliva, A. Sun database: Exploring a large collection of scene categories. *International Journal of Computer Vision (IJCV)*, 2016.

Yadav, P., Tam, D., Choshen, L., Raffel, C. A., and Bansal, M. Ties-merging: Resolving interference when merging models. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 7093–7115. Curran Associates, Inc., 2023.

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

Yu, L., Yu, B., Yu, H., Huang, F., and Li, Y. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*, 2024.

Zhang, Y., Baldridge, J., and He, L. PAWS: Paraphrase Adversaries from Word Scrambling. In *Proc. of NAACL*, 2019.

Zhou, J., Lu, T., Mishra, S., Brahma, S., Basu, S., Luan, Y., Zhou, D., and Hou, L. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

A. Experimental Setup

Datasets & Models. For vision tasks, we follow Ilharco et al. (2021) and fine-tune ViT-B/16, ViT-B/32, and ViT-L/14 models on eight image classification datasets: Cars (Krause et al., 2013), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019), GTSRB (Stallkamp et al., 2011), MNIST (LeCun, 1998), RESISC45 (Cheng et al., 2017), SUN397 (Xiao et al., 2016), and SVHN (Netzer et al., 2011). For language tasks, we fine-tune T5-Base and T5-Large (Raffel et al., 2020) on seven multiple-choice datasets: QASC (Khot et al., 2020), WikiQA (Cohen et al., 2018), QuaRTz (Tafjord et al., 2019), PAWS (Zhang et al., 2019), Story Cloze (Sharma et al., 2018), Winogrande (Sakaguchi et al., 2020), and WSC (Levesque et al., 2012). For reasoning tasks, we evaluate merging models trained via reinforcement learning with verifiable rewards, using OLMo-3-7B (Olmo et al., 2025) as the base model. Specifically, we merge three publicly available RL-Zero checkpoints trained on math, code, and instruction-following. Following the evaluation protocol in Olmo et al. (2025), we evaluate mathematical reasoning on AIME 2024 & 2025. Meanwhile, coding and instruction-following abilities are evaluated using HumanEval (Chen et al., 2021), HumanEval+ (Liu et al., 2023) and IFEval (Zhou et al., 2023).

Implementation Details. Following Yadav et al. (2023), for vision experiments we only fine-tune and merge the vision encoder. For language experiments, embedding layers are always averaged. Following Jin et al. (2023), only 2D weight matrices are merged via each merging method's respective rule and all other parameters are averaged. In LoRA (Hu et al., 2022) experiments, we follow the setup of Stoica et al. (2025) and use rank-16 adapters on all linear layers. Methods that rely on auxiliary data use the validation splits of datasets. In the data-free setting, Task Arithmetic uses a scaling factor of $\alpha = 0.4$ following Yadav et al. (2023), while Iso-C and TSV use $\alpha = 1$.

B. Theorem Proofs

B.1. Layer-wise Interference Minimization

Lemma B.1. *Let $\boldsymbol{W}_t \in \mathbb{R}^{D_o \times D_i}$ and $\boldsymbol{C}_t \in \mathbb{R}^{D_i \times D_i}$ for $t = 1 \dots T$. Define $\boldsymbol{A} := \sum_{t=1}^{T} \boldsymbol{C}_t \in \mathbb{R}^{D_i \times D_i}$ and $\boldsymbol{B} := \sum_{t=1}^{T} \boldsymbol{W}_t \boldsymbol{C}_t \in \mathbb{R}^{D_o \times D_i}$. If the matrices $\boldsymbol{C}_t$ are symmetric positive semidefinite (denoted from now on by $\boldsymbol{C}_t \succeq 0$), then the matrix equation $\boldsymbol{W}\boldsymbol{A} = \boldsymbol{B}$ always admits at least one solution, and the set of solutions is*

$$\left\{ \boldsymbol{W}^\star + \boldsymbol{Z}\left(\mathbb{I}_{D_i} - \boldsymbol{A}\boldsymbol{A}^\dagger\right) \mid \boldsymbol{Z} \in \mathbb{R}^{D_o \times D_i} \right\}, \tag{6}$$

with $\boldsymbol{W}^\star = \boldsymbol{B}\boldsymbol{A}^\dagger$ being the minimum Frobenius norm solution. Moreover, the solution is unique iff $\boldsymbol{A} = \sum_{t=1}^{T} \boldsymbol{C}_t$ is invertible, which is equivalent to $\ker\left(\sum_{t=1}^{T} \boldsymbol{C}_t\right) = \bigcap_{t=1}^{T} \ker(\boldsymbol{C}_t) = \{0\}$.

Proof. A solution to the equation $\boldsymbol{W}\boldsymbol{A} = \boldsymbol{B}$ exists if and only if each row of $\boldsymbol{B}$ belongs to the row space of $\boldsymbol{A}$, i.e. $\boldsymbol{B} = \boldsymbol{B}\boldsymbol{A}^\dagger\boldsymbol{A}$. We solve the equation row by row. Let $\boldsymbol{w} \in \mathbb{R}^{D_i}$ be a row of $\boldsymbol{W} \in \mathbb{R}^{D_o \times D_i}$ and $\boldsymbol{b} \in \mathbb{R}^{D_i}$ the corresponding row of $\boldsymbol{B}$. The equation $\boldsymbol{W}\boldsymbol{A} = \boldsymbol{B}$ is equivalent to $\boldsymbol{A}\boldsymbol{w} = \boldsymbol{b}$ for all $(\boldsymbol{w}, \boldsymbol{b})$. Thus, the existence of a solution is equivalent to $\boldsymbol{b} \in \mathrm{Im}(\boldsymbol{A}) = \ker(\boldsymbol{A})^\perp$ for all row $\boldsymbol{b}$ of $\boldsymbol{B}$, where we use the fact that $\mathrm{Im}(\boldsymbol{A}) = \ker(\boldsymbol{A})^\perp$ since $\boldsymbol{A}$ is symmetric. Therefore, $\boldsymbol{b} \in \mathrm{Im}(\boldsymbol{A})$ if and only if $\boldsymbol{x}^\top \boldsymbol{b} = 0$ for all $\boldsymbol{x} \in \ker(\boldsymbol{A})$. We now prove this condition, which also reads: $\boldsymbol{B}\boldsymbol{x} = 0$ for all $\boldsymbol{x} \in \ker(\boldsymbol{A})$.

Let $\boldsymbol{x} \in \ker(\boldsymbol{A})$, so that $\boldsymbol{A}\boldsymbol{x} = 0$. Then $0 = \boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x} = \sum_{t=1}^{T} \boldsymbol{x}^\top \boldsymbol{C}_t \boldsymbol{x}$, which is equivalent to $\boldsymbol{x}^\top C_t \boldsymbol{x} = 0 \ \forall t$ since $\boldsymbol{x}^\top \boldsymbol{C}_t \boldsymbol{x} \geq 0 \forall t$ (by the positive semidefinitness of each $\boldsymbol{C}_t$). Since each $\boldsymbol{C}_t \succeq 0$, there exists $\boldsymbol{C}_t^{1/2}$ such that $\boldsymbol{C}_t = \boldsymbol{C}_t^{1/2}\boldsymbol{C}_t^{1/2}$. Thus

$$\boldsymbol{x}^\top \boldsymbol{C}_t \boldsymbol{x} = 0 \quad \forall t \iff \|\boldsymbol{C}_t^{1/2}\boldsymbol{x}\|^2 = 0 \quad \forall t \tag{7}$$

$$\iff \boldsymbol{C}_t^{1/2}\boldsymbol{x} = 0 \quad \forall t \tag{8}$$

$$\implies \boldsymbol{C}_t \boldsymbol{x} = 0 \quad \forall t, \tag{9}$$

So

$$\boldsymbol{B}\boldsymbol{x} = \sum_{t=1}^{T} \boldsymbol{W}_t \boldsymbol{C}_t \boldsymbol{x} = 0. \tag{10}$$

Therefore, the system $\boldsymbol{W}\boldsymbol{A} = \boldsymbol{B}$ admits a solution. It is easy to check that all solutions are of the form

$$\boldsymbol{W} = \boldsymbol{W}^\star + \boldsymbol{Z}\left(\mathbb{I}_{D_i} - \boldsymbol{A}\boldsymbol{A}^\dagger\right) \ \forall \boldsymbol{Z} \in \mathbb{R}^{D_o \times D_i}. \tag{11}$$

So the solution is unique iff the free term always vanishes, that is, iff $\boldsymbol{A}\boldsymbol{A}^\dagger = \mathbb{I}_{D_i}$. Since $\boldsymbol{A}\boldsymbol{A}^\dagger$ is the orthogonal projector onto $\mathrm{Im}(\boldsymbol{A})$, this happens iff $\boldsymbol{A}$ is invertible, or equivalently, if $\boldsymbol{A}$ is positive definite (since $\boldsymbol{A} \succeq 0$ as a sum of positive semidefinite matrices), or equivalently, if $\{0\} = \ker\left(\sum_{t=1}^T \boldsymbol{C}_t\right) = \bigcap_{t=1}^T \ker(\boldsymbol{C}_t)$. □

Lemma B.2. *Let $\boldsymbol{W}_t \in \mathbb{R}^{D_o \times D_i}$ for $t = 1 \dots T$. Define $\boldsymbol{C}_t = \mathbb{E}_{\boldsymbol{z}\sim\mathcal{D}_t}[\boldsymbol{z}\boldsymbol{z}^\top] \in \mathbb{R}^{D_i \times D_i}$ for each $t = 1 \dots T$, where $\boldsymbol{z}$ denote a D_i-dimensional random vector distributed according to $\mathcal{D}_t$. Then the matrix $\boldsymbol{W}^\star \in \mathbb{R}^{D_o \times D_i}$ defined by*

$$\boldsymbol{W}^\star = \left(\sum_{t=1}^T \boldsymbol{W}_t \boldsymbol{C}_t\right)\left(\sum_{t=1}^T \boldsymbol{C}_t\right)^\dagger, \tag{12}$$

is the minimum Frobenius norm solution to the problem

$$\min_{\boldsymbol{W}} g(\boldsymbol{W}), \quad g(\boldsymbol{W}) = \sum_{t=1}^T \mathbb{E}_{\boldsymbol{z}\sim\mathcal{D}_t}\left[\|\boldsymbol{W}\boldsymbol{z} - \boldsymbol{W}_t\boldsymbol{z}\|_2^2\right]. \tag{13}$$

Proof. Expanding $g(\boldsymbol{W})$ and using the cyclic property of the trace, we get

$$g(\boldsymbol{W}) = \sum_{t=1}^T \mathbb{E}_{\boldsymbol{z}\sim\mathcal{D}_t}\left[\boldsymbol{z}^\top(\boldsymbol{W} - \boldsymbol{W}_t)^\top(\boldsymbol{W} - \boldsymbol{W}_t)\boldsymbol{z}\right] \tag{14}$$

$$= \sum_{t=1}^T \mathbb{E}_{\boldsymbol{z}\sim\mathcal{D}_t}\left((\boldsymbol{W} - \boldsymbol{W}_t)^\top(\boldsymbol{W} - \boldsymbol{W}_t)\boldsymbol{z}\boldsymbol{z}^\top\right) \tag{15}$$

$$= \sum_{t=1}^T \left((\boldsymbol{W} - \boldsymbol{W}_t)^\top(\boldsymbol{W} - \boldsymbol{W}_t)\boldsymbol{C}_t\right). \tag{16}$$

So

$$\nabla_{\boldsymbol{W}} g(\boldsymbol{W}) = \sum_{t=1}^T (\boldsymbol{W} - \boldsymbol{W}_t)(\boldsymbol{C}_t + \boldsymbol{C}_t^\top) = 2\sum_{t=1}^T (\boldsymbol{W} - \boldsymbol{W}_t)\boldsymbol{C}_t, \tag{17}$$

where the last equality uses the fact that $\boldsymbol{C}_t$ is symmetric. The function g is convex and differentiable, and therefore its minimizers correspond to the solution of the equation $\nabla_{\boldsymbol{W}} g(\boldsymbol{W}) = 0$, which is equivalent to $\boldsymbol{W}\boldsymbol{A} = \boldsymbol{B}$ with $\boldsymbol{A} = \sum_{t=1}^T \boldsymbol{C}_t \in \mathbb{R}^{D_i \times D_i}$ and $\boldsymbol{B} = \sum_{t=1}^T \boldsymbol{W}_t\boldsymbol{C}_t \in \mathbb{R}^{D_o \times D_i}$. By Lemma B.1, this equation always admits a solution since each $\boldsymbol{C}_t$ is a covariance matrix, and thus symmetric positive semidefinite. The minimum-Frobenius-norm solution is

$$\boldsymbol{W} = \boldsymbol{B}\boldsymbol{A}^\dagger = \left(\sum_{t=1}^T \boldsymbol{W}_t\boldsymbol{C}_t\right)\left(\sum_{t=1}^T \boldsymbol{C}_t\right)^\dagger = \boldsymbol{W}^\star \tag{18}$$

□

B.2. Covariance Estimation

Theorem 3.1 (Covariance Estimation)**.** *Consider a linear layer fine-tuned using gradient descent for K iterations with learning rate η, and let $\boldsymbol{z}^{(k)}$, $\boldsymbol{g}^{(k)}$ denote the layer's input and its output gradient at iteration k, respectively. Define*

$$\overline{\boldsymbol{G}} := \textstyle\sum_{k=0}^K \mathbb{E}\left[\boldsymbol{g}^{(k)}\boldsymbol{z}^{(k)\top}\right],$$
$$\overline{\boldsymbol{S}} := \textstyle\sum_{k=0}^K \mathbb{E}\left[\boldsymbol{z}^{(k)}\boldsymbol{z}^{(k)\top}\|\boldsymbol{g}^{(k)}\|^2\right],$$
$$\widetilde{\boldsymbol{S}} := \textstyle\sum_{k=0}^K \mathbb{E}\left[\boldsymbol{z}^{(k)}\boldsymbol{z}^{(k)\top}\right]\mathbb{E}\left[\|\boldsymbol{g}^{(k)}\|^2\right],$$

where the expectation is taken over $\mathcal{D}_t$, and $\|\cdot\|$ denotes the Euclidean norm. Then, the angular distance between $\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t$ and the final covariance $\boldsymbol{C}_t^{(K)}$ satisfies

$$\measuredangle\left(\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t,\ \boldsymbol{C}_t^{(K)}\right) \le \epsilon^{(\mathrm{cross})} + \epsilon^{(\mathrm{corr})} + \epsilon^{(\mathrm{drift})}, \tag{5}$$

where $\epsilon^{(\mathrm{cross})} = \measuredangle\left(\overline{\boldsymbol{G}}^\top\overline{\boldsymbol{G}},\ \overline{\boldsymbol{S}}\right)$ is the cross-term error, $\epsilon^{(\mathrm{corr})} = \measuredangle\left(\overline{\boldsymbol{S}},\ \widetilde{\boldsymbol{S}}\right)$ is the correlation error, and $\epsilon^{(\mathrm{drift})} = \measuredangle\left(\widetilde{\boldsymbol{S}},\ \boldsymbol{C}_t^{(K)}\right)$ is the drift error. In particular, $\boldsymbol{\Delta}_t^\top\boldsymbol{\Delta}_t \propto \boldsymbol{C}_t^{(K)}$ when all three errors vanish.

Proof. After K iterations of full-batch gradient descent with learning rate η, the difference matrix can be written as

$$\boldsymbol{\Delta}_t = \boldsymbol{W}^{(K+1)} - \boldsymbol{W}^{(0)} = -\eta \sum_{k=0}^{K} \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_t}\left[\boldsymbol{g}^{(k)}\boldsymbol{z}^{(k)\top}\right] = -\eta\,\overline{\boldsymbol{G}},$$

and therefore $\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t = \eta^2\,\overline{\boldsymbol{G}}^\top \overline{\boldsymbol{G}}$. Successively applying the triangle inequality for angular distance,

$$\theta\big(\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t,\ \boldsymbol{C}_t^{(K)}\big) \ \le\ \theta\big(\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t,\ \overline{\boldsymbol{G}}^\top \overline{\boldsymbol{G}}\big) \ +\ \theta\big(\overline{\boldsymbol{G}}^\top \overline{\boldsymbol{G}},\ \overline{\boldsymbol{S}}\big) \ +\ \theta\big(\overline{\boldsymbol{S}},\ \widetilde{\boldsymbol{S}}\big) \ +\ \theta\big(\widetilde{\boldsymbol{S}},\ \boldsymbol{C}_t^{(K)}\big).$$

Since $\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t$ and $\overline{\boldsymbol{G}}^\top \overline{\boldsymbol{G}}$ are collinear, the first term vanishes, giving

$$\theta\big(\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t,\ \boldsymbol{C}_t^{(K)}\big) \ \le\ \epsilon^{(\mathrm{cross})} + \epsilon^{(\mathrm{corr})} + \epsilon^{(\mathrm{drift})}.$$

When all three errors vanish, $\measuredangle(\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t,\ \boldsymbol{C}_t^{(K)}) = 0$, which implies $\boldsymbol{\Delta}_t^\top \boldsymbol{\Delta}_t \propto \boldsymbol{C}_t^{(K)}$. □

C. Full Results on Vision and Language Experiments

In this section, we report the results of Figure 3 in tabular form. Table 2 reports test accuracies under full fine-tuning, while Table 3 reports the corresponding results when models are fine-tuned with LoRA (Hu et al., 2022).

Method (↓)	**Data-free**	**NLP**		**Vision**		
Model (→)		**T5-B**	**T5-L**	**ViT-B/16**	**ViT-B/32**	**ViT-L/14**
Zeroshot	-	54.0	51.3	55.5	48.2	65.2
Experts	-	76.0	82.4	94.6	90.4	94.1
RegMean	✗	74.5	80.8	87.6	83.0	90.0
TA	✗	71.4	69.7	76.1	69.7	84.3
TA	✓	60.5	59.8	24.3	26.5	41.8
Average	✓	64.8	50.9	72.2	65.5	79.3
Iso-C	✓	66.1	57.6	88.4	81.6	90.5
TSV	✓	72.9	74.7	88.8	**83.3**	91.1
ACTMat	✓	**76.0**	**79.8**	**89.5**	82.9	**92.2**

Table 2. Comparison between merging methods across multiple settings (NLP models fine-tuned on 7 tasks and vision models fine-tuned on 8 tasks). All model parameters are fine-tuned. The ✓symbol indicates that no data is used by the method. Results are reported on test sets.

Method (↓)	**Data-free**	**NLP**		**Vision**		
Model (→)		**T5-B**	**T5-L**	**ViT-B/16**	**ViT-B/32**	**ViT-L/14**
Experts	-	76.5	79.2	84.1	80.7	89.0
Zeroshot	-	54.0	51.3	55.5	52.0	65.2
RegMean	✗	**75.9**	**77.7**	**80.3**	**76.5**	**86.9**
TA	✗	66.7	70.9	75.7	70.5	83.0
TA	✓	54.2	55.0	66.1	53.5	77.6
Iso-C	✓	56.1	52.1	57.5	50.4	66.8
Average	✓	62.2	57.2	64.3	56.8	72.4
K-Iso-C	✓	64.7	69.4	74.9	70.4	83.6
K-TSV	✓	67.6	71.1	74.8	69.6	83.4
TSV	✓	**72.2**	75.4	78.4	73.0	85.4
ACTMat	✓	72.0	**76.7**	**78.6**	**74.6**	**86.1**

Table 3. Comparison between merging methods across multiple settings (NLP models fine-tuned on 7 tasks and vision models fine-tuned on 8 tasks). Model parameters are fine-tuned using **LoRA**. The ✓symbol indicates that no data is used by the method. Results are reported on test sets.