ALIGN-SAM: SEEKING FLATTER MINIMA FOR BETTER CROSS-SUBSET ALIGNMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Sharpness-Aware Minimization (SAM) has proven effective in enhancing deep neural network performance by simultaneously minimizing the training loss and the sharpness of the loss landscape, thereby guiding models toward flatter minima that are empirically linked to improved generalization. From another perspective, generalization can be seen as a model's ability to remain stable under distributional variability. In particular, effective learning requires that updates derived from different subsets or resamplings of the same data distribution remain consistent. In this work, we investigate the connection between the flatness induced by SAM and the alignment of gradients across random subsets of the data distribution, and propose *Align-SAM* as a novel strategy to further enhance model generalization. Align-SAM extends the core principles of SAM by promoting optimization toward flatter minima on a primary subset (the training set), while simultaneously enforcing low loss on an auxiliary subset that is drawn from the same distribution. This dual-objective approach leads to solutions that are not only resilient to local perturbations but also robust against distributional shifts in each training iteration. Empirical evaluations demonstrate that Align-SAM consistently improves generalization across diverse datasets and challenging settings, including scenarios with noisy labels and limited data availability.

1 INTRODUCTION

Deep neural networks have emerged as the dominant approach for solving complex tasks such as classification, often outperforming traditional machine learning models. These models learn by adjusting a vast number of parameters to minimize prediction errors or maximize task-specific performance. In practice, training is conducted on a finite dataset $\mathcal{S}$, sampled from an unknown underlying distribution $\mathcal{D}$. The quality and alignment of this dataset with the target distribution significantly impact model efficiency and performance Hestness et al. (2017); Kaplan et al. (2020). Despite their ability to learn complex patterns, deep learning models can also capture noise or random fluctuations in training data, leading to overfitting Arpit et al. (2017); Zhang et al. (2016); Liu et al. (2020). This results in excellent performance on training data but poor predictions on new, unseen data, especially with domain shifts. Generalization McAllester (1999); Dziugaite & Roy (2017b), measured by comparing prediction errors on S and $\mathcal{D}$, becomes crucial. Balancing a model's ability to fit training data with its risk of overfitting is a key challenge in machine learning.

Several studies have been done on this problem, both from theoretical and practical perspectives. Statistical learning theory has proposed different complexity measures that are capable of controlling generalization errors (Vapnik, 1998; Bartlett & Mendelson, 2003; Mukherjee et al., 2002; Bousquet & Elisseeff, 2002; Poggio et al., 2004). In general, they develop a bound for the general error on $\mathcal{D}$. Theory suggests that minimizing the intractable general error on $\mathcal{D}$ is equivalent to minimizing the empirical loss on S with some constraints to the complexity of models and training size (Alquier et al., 2016b). An alternative strategy for mitigating generalization errors involves the utilization of an optimizer to learn optimal parameters for models with a specific local geometry. This approach enables models to find wider local minima (*i.e.*, flat minima), which makes them more robust against data shift between training and testing sets (Jiang et al., 2020; Petzka et al., 2021; Huang et al., 2025).

The connection between a model's generalization and the width of minima has been investigated theoretically and empirically in many studies, notably (Hochreiter & Schmidhuber, 1994; Neyshabur et al., 2017; Dinh et al., 2017; Fort & Ganguli, 2019). A specific method within this paradigm is Sharpness-Aware Minimisation (SAM) (Foret et al., 2021), which has emerged as an effective technique for enhancing the generalization ability of deep learning models. SAM seeks a perturbed model within the vicinity of a current model that maximizes the loss over a training set. Eventually, SAM leads the model to the region where both the current model and its perturbation model have low loss values, ensuring flatness. The success of SAM and its variants (Kwon et al., 2021; Kim et al., 2022) has inspired further investigation into its formulation and behavior, as evidenced by recent works such as (Kaddour et al., 2022; Möllenhoff & Khan, 2022; Andriushchenko & Flammarion, 2022a; Ji et al., 2024).

Additionally, inspired by the PAC-Bayes theorem Alquier (2023), SAM provides an upper bound on the generalization loss over the data distribution $\mathcal{D}$ by considering the loss of perturbed models trained on a random dataset $S \sim \mathcal{D}$. This guides optimization toward minimizing the worst-case loss within a neighborhood of parameters. Such a framework has been shown to encourage convergence to flatter minima on the random dataset S, which in turn promotes improving generalization. In practice, however, directly minimizing sharpness over the entire training set S would require forward and backward passes on all of S, which is computationally infeasible with large datasets. Instead, standard stochastic optimizers, such as SGD or Adam, are applied that rely on randomly sampled mini-batches, thereby reducing sharpness on each subset and stochastically approximating minimization of sharpness over the full dataset S.

From a complementary perspective, generalization can be viewed as the model's ability to remain reliable on new subsets drawn from the same data distribution $\mathcal{D}$. Specifically, we adopt the viewpoint that a model exhibits strong generalization if, although optimized primarily on one random subset S, it can also perform well on another independently drawn auxiliary subset S^a, where $S, S^a \sim \mathcal{D}$. Motivated by this, we formulate our objective as finding models that minimize sharpness and loss on the primary subset while simultaneously maintaining low loss on the auxiliary subset, thereby ensuring stability across resamplings of the data distribution. To achieve this, we propose *Align-SAM*, a novel method that updates model parameters toward solutions that are both flat and low-loss on the primary subset, while maintaining robust performance across auxiliary subsets, thus implicitly promoting stronger generalization to the full distribution.

In summary, our contributions in this work are as follows:

- We approach generalization from a novel perspective by framing it as an alignment across random subsets drawn from the same data distribution. Building on this viewpoint, we propose Align-SAM, a method designed to enhance both model flatness and stability under distributional variability. Align-SAM primarily updates model parameters by guiding them toward regions in parameter space that minimize sharpness and loss on a primary subset, while simultaneously encouraging strong performance on an auxiliary subset sampled independently from the same distribution. This is achieved by leveraging a combination of gradients computed on both subsets during optimization.
- We demonstrate the effectiveness of Align-SAM in enhancing generalization performance across a variety of settings. Our evaluation begins with image classification tasks, covering both training from scratch and transfer learning on datasets ranging from small to large scale. We further assess its robustness under noisy label conditions with varying noise levels. Additionally, we extend our experiments to meta-learning scenarios to evaluate Align-SAM's ability to generalize beyond meta-training tasks and adapt across diverse domains. The consistent performance gains across these experiments show that Align-SAM not only improves robustness to label noise and generalization across tasks but also promotes more stable and reliable predictions in varied settings.

2 RELATED WORKS

Sharpness-Aware Minimization. The correlation between the wider minima and the generalization capacity has been extensively explored both theoretically and empirically in various studies Tan et al. (2025); Jiang et al. (2020); Petzka et al. (2021); Dziugaite & Roy (2017a); Zhuang et al. (2022); Kwon et al. (2021). Many works suggested that finding flat minimizers might help to reduce generalization error and increase robustness to data distributional shift problems in various settings Jiang et al.

(2020); Petzka et al. (2021); Huang et al. (2025). There are multiple works have explored the impact of different training parameters, including batch size, learning rate, gradient covariance, and dropout, on the flatness of discovered minima such as Keskar et al. (2017); Jastrzebski et al. (2017); Wei et al. (2020); Deng et al. (2025).

Sharpness-aware minimization (SAM) (Foret et al., 2021) is a recent optimization technique designed to improve the generalization error of neural networks by considering the sharpness of the loss landscape during training. SAM minimizes the worst-case loss around the current model and effectively updates models towards flatter minima to achieve low training loss and maximize generalization performance on new and unseen data. SAM has been successfully applied to various tasks and domains, such as vision models (Chen et al., 2021), language models (Bahri et al., 2022), federated learning (Qu et al., 2022; Xing et al., 2025), Bayesian Neural Networks (Nguyen et al., 2023), domain generalization (Cha et al., 2021), multi-task learning (Phan et al., 2022) and meta-learning bi-level optimization (Abbas et al., 2022). Multiple varieties of SAM have been developed to address limitations of the original method, including ASAM (Kwon et al., 2021), Friendly-SAM (Li et al., 2024), GSAM (Du et al., 2022), VASSO (Li & Giannakis, 2024), and other curvature- or alignment-aware extensions. Efficiency-oriented approaches such as SAF Du et al. (2022) approximate SAM's perturbation to reduce computational overhead, GNAM Zhang et al. (2023) promotes first-order flatness through gradient-norm regularization. LookSAM Liu et al. (2022) further improves efficiency by reusing the same perturbation direction across k consecutive iterations, significantly lowering SAM's computation while maintaining its flatness-seeking behaviour. Recent work has also been inspired by the Lookahead optimizer to deploy multi-step strategies to explore flatter regions of the loss landscape: Lookahead-SAM (Yu et al., 2024) integrates Lookahead's extrapolation–interpolation mechanism with SAM to reach wider minima, whereas Lookbehind-SAM (Mordido et al., 2024) incorporates backward steps before the SAM update to better navigate sharp regions.

Implicit Biases and Behaviors of SAM. SAM was inspired by the PAC-Bayes theorem Alquier et al. (2016a); Alquier (2023); Alquier et al. (2016b), which provides an upper bound on generalization loss and motivates the pursuit of flat minima. Several works have since sought to better understand and improve Sharpness-Aware Minimization (SAM). Andriushchenko and Flammarion Andriushchenko & Flammarion (2022b) offer theoretical insights into SAM's optimization dynamics, emphasizing its implicit regularization effects. Compagnoni et al. Monzio Compagnoni et al. (2023) model SAM using a stochastic differential equation (SDE), providing a continuous-time interpretation of its behavior. Wen et al. (Wen et al., 2023b) study how SAM reduces sharpness during optimization, while Chen et al. Chen et al. (2023) show that SAM generalizes better than SGD by avoiding sharp minima. To further improve SAM, Luo et al. (Luo et al., 2024) introduce explicit eigenvalue regularization to control curvature during training. Finally, Wen et al. (Wen et al., 2023a) argue that the generalization benefits of sharpness-aware algorithms stem not only from sharpness minimization but also from other favorable inductive biases.

A complementary strand of work examines data distribution, class imbalance, and long-tailed regimes, where sharpness varies significantly across classes. Nguyen et al. (2024) analyze the features learned under SAM and show that SAM encourages models to rely on both simple and complex features. They further demonstrate that modifying the training data distribution to reduce simplicity bias improves in-distribution generalization. In class-imbalanced settings, several methods adapt SAM to better handle head–tail disparities. ImbSAM Zhou et al. (2023a) incorporates imbalance-aware perturbations to prevent minority classes from being overshadowed by head-class gradients. Class-Conditional SAM Zhou et al. (2023b) aligns SAM's perturbation with class-specific curvature to reduce sharpness for tail classes, improving long-tailed accuracy. Focal-SAM Li et al. (2025) integrates focal reweighting with SAM to emphasize hard or misclassified instances while maintaining stable optimization.

3 Proposed Method

Notions. We start by introducing the notions used throughout our paper. We denote $\mathcal{D}$ as the data/label distribution to generate pairs of data/label (x, y). Given a model with the model parameter θ, we denote the per-sample loss induced by (x, y) as $\ell(x, y; \theta)$. Let S be a random subset drawn from the distribution $\mathcal{D}$. We denote the *empirical* and *generalization* losses as $\mathcal{L}_S(\theta) = \mathbb{E}_S[\ell(x, y; \theta)]$ and $\mathcal{L}_\mathcal{D}(\theta) = \mathbb{E}_\mathcal{D}[\ell(x, y; \theta)]$ respectively. We define $\mathcal{L}_\mathcal{D}(\theta \mid S)$ as an *upper bound defined over S* of the general loss $\mathcal{L}_{\mathcal{D}}\left(\theta\right)$. Note that inspired by SAM (Foret et al., 2021), we use the sharpness over S to define $\mathcal{L}_{\mathcal{D}}\left(\theta \mid S\right)$ (see Theorem 1). Finally, we use $|A|$ to denote the cardinality of a set A.

3.1 Problem Formulation

Given a random subset S^t whose examples are sampled from $\mathcal{D}$ (*i.e.*, $S^t \sim \mathcal{D}^{N_t}$ with $N_t = |S^t|$), we use $\mathcal{L}_{\mathcal{D}}\left(\theta \mid S^t\right)$ to train models. S^t is known as the training set. Among the models that minimize this loss, we select the one that minimizes the general loss as follows:

$$\min_{\theta^*} \mathcal{L}_{\mathcal{D}}\left(\theta^*\right) \text{ s.t. } \theta^* \in \mathcal{A}_{\mathcal{D}}\left(S^t\right) := \text{argmin}_{\theta} \mathcal{L}_{\mathcal{D}}\left(\theta \mid S^t\right). \tag{1}$$

We note that $\mathcal{A}_{\mathcal{D}}\left(S^t\right)$ returns the optimal models θ^* that minimizes the upper bound $\mathcal{L}_{\mathcal{D}}\left(\theta \mid S^t\right)$. Among the set of such minimizers θ^*, we select the one that further minimizes the true generalization loss $\mathcal{L}_{\mathcal{D}}$. The reason for the formulation in (1) is that although $\mathcal{L}_{\mathcal{D}}\left(\theta \mid S^t\right)$ is an upper bound of the general loss $\mathcal{L}_{\mathcal{D}}\left(\theta\right)$, there always exists a gap between them. Therefore, the additional outer minimization helps to refine the solutions.

We now denote S^a (*i.e.*, $S^a \sim \mathcal{D}^{N_a}$ with $N_a = |S^a|$) as an other random subset sampled from $\mathcal{D}$, S^a is called the auxiliary set. With respect to this auxiliary set, we have the following theorem.

Theorem 1. *Under conditions $L_D(\theta) \leq E_{\epsilon_i \sim N(0,\rho)} L_D(\theta + \epsilon)$ similar to SAM (Foret et al., 2021), with a probability greater than $1 - \delta$ (i.e., $\delta \in [0, 1]$) over the choice of $S^a \sim \mathcal{D}^{N_a}$, we then have for any optimal models $\theta^* \in \mathcal{A}_{\mathcal{D}}(S^t)$:*

$$\mathcal{L}_{\mathcal{D}}\left(\theta^*\right) \leq \mathcal{L}_{\mathcal{D}}\left(\theta^* \mid S^a\right) + \frac{8L}{\sqrt{N_a}}\sqrt{\log\frac{N_a + k}{\delta}} + \frac{4L}{\sqrt{N_a}}O(1) + \frac{4L}{\sqrt{N_a}} k \log\left(1 + \frac{\|\theta^*\|^2}{\rho}\left(1 + \sqrt{\log N_a/k}\right)\right), \tag{2}$$

*where we denote $\mathcal{L}_{\mathcal{D}}\left(\theta^* \mid S\right) := \max_{\theta' : \|\theta' - \theta^*\|_2 \leq \rho} \mathcal{L}_S\left(\theta'\right)$ for any random subset $S \sim \mathcal{D}^N$ (i.e S^t, S^a), and L is the upper bound of the loss function (*i.e.*, $\ell\left(x, y; \theta\right) \leq L, \forall x, y, \theta$), k is the model size as the length of vector θ, and $\rho > 0$ is the perturbation radius.*

According to Theorem 1, $\mathcal{L}_{\mathcal{D}}\left(\theta^* \mid S\right) := \max_{\theta' : \|\theta' - \theta^*\|_2 \leq \rho} \mathcal{L}_S\left(\theta'\right)$ can be viewed as an upper bound of the generalization loss $\mathcal{L}_{\mathcal{D}}(\theta^*)$, up to a constant difference. Moreover, our theorem 1 (see Appendix A.1 for proof) can be viewed as an extension of Theorem 1 in Foret et al. (2021), where we apply the PAC-Bayes theorem from Alquier et al. (2016a) to prove an upper bound for the generalization loss of any bounded loss, instead of the 0-1 loss in Foret et al. (2021). We can generalize this proof for S^t to explain why we use $\mathcal{L}_{\mathcal{D}}\left(\theta \mid S^t\right) := \max_{\theta' : \|\theta' - \theta\|_2 \leq \rho} \mathcal{L}_{S^t}\left(\theta'\right)$ as an objective to minimize, as in (1).

Based on Theorem 1, we can rewrite the objectives in (1) as:

$$\min_{\theta^*} \mathcal{L}_{\mathcal{D}}\left(\theta^* \mid S^a\right) \text{ s.t. } \theta^* \in \mathcal{A}_{\mathcal{D}}\left(S^t\right) := \text{argmin}_{\theta} \, \mathcal{L}_{\mathcal{D}}\left(\theta \mid S^t\right), \tag{3}$$

where $\mathcal{L}_{\mathcal{D}}\left(\theta \mid S\right) := \max_{\theta' : \|\theta' - \theta\|_2 \leq \rho} \mathcal{L}_S\left(\theta'\right)$. Among all models that minimize the upper generalization bound on a random training subset S^t, we select the one that further minimizes the upper generalization bound on an independently drawn auxiliary subset S^a. In other words, the optimal solution is the one that achieves low sharpness and loss on the primary subset while simultaneously maintaining low loss on the auxiliary subset.

Our theory works for $S^t, S^a \sim \mathcal{D}$, where $\mathcal{D}$ is the distribution to generate (x, y). In the practical version of Algorithm 1, we replace $\mathcal{D}$ by the empirical distribution S and at each iteration, we sample two mini-batches $B^t, B^a \sim S$. Because when the training size (i.e. $|S|$) approaches ∞, the distribution S asymptotically converges to the distribution $\mathcal{D}$. Using stochastic optimization, we reformulate this objective into an iterative update scheme, where the model is trained with two independently drawn mini-batches, B^t and B^a, such that each update encourages convergence toward flat minima while aligning performance across subsets in every iteration.

3.2 Our Solution

Our motivation here is to primarily optimize the loss over the training set S^t, while using S^a to further enhance the generalization ability, where both S^t and S^a are random subsets drawn from the same data distribution. Our formulation in (3) has the same form as a bi-level optimization problem similar to MAML (Finn et al., 2017), developed for meta-learning. Inspired by MAML, a naive approach would be to consider $f(\theta) := \text{argmin}_\theta \mathcal{L}_\mathcal{D}\left(\theta \mid S^t\right)$ (*i.e.*, $f(\theta) = \theta - \eta \nabla_\theta \mathcal{L}_\mathcal{D}\left(\theta \mid S^t\right)$) and finding $\theta^* := \text{argmin}_\theta \mathcal{L}_\mathcal{D}\left(f(\theta) \mid S^a\right)$ with respect to θ. However, this naive approach *does not align* with our objective, as it mainly focuses on optimizing the loss $\mathcal{L}_\mathcal{D}\left(f(\theta) \mid S^a\right)$ over the auxiliary set S^a, in which, the auxiliary set acts like the validation set in MAML. Here we note that in Franceschi et al. (2018), bi-level optimization was employed to learn optimal hyperparameters (e.g., the weight of a regularizer) by finding hyperparameters such that a model trained on a training set performs well on a validation set. This is fundamentally different from our aim, which is to study how to achieve flat minima that align two independent random subsets in every update step. Moreover, directly adapting the MAML bi-level formulation would require distinct training and validation sets, which are often unavailable in most scenarios. For these reasons, both our theoretical framework and technical approach differ substantially from those in Franceschi et al. (2018).

Using stochastic optimization, we reformulate the objective (3) into an iterative update scheme, where the model is trained using two independently drawn mini-batches B^t and B^a, with both batches sampled independently from the training set S. The *Align-SAM* is presented as follows: *at each iteration, our primary objective is to optimize $\mathcal{L}\left(\theta \mid B^t\right)$, primarily based on its gradients, in such a way that future models are able to implicitly perform well on B^a.* To achieve this, similar to SAM (Foret et al., 2021), we approximate $\mathcal{L}\left(\theta \mid B^t\right) = \max_{\|\theta' - \theta\| \le \rho} \mathcal{L}_{B^t}\left(\theta'\right) \approx \mathcal{L}_{B^t}\left(\theta + \eta_1 \nabla \mathcal{L}_{B^t}\left(\theta\right)\right)$ for a sufficient small learning rate $\eta_1 > 0$ (i.e., $\eta_1 \|\nabla \mathcal{L}_{B^t}\left(\theta\right)\| \le \rho$) and $\mathcal{L}\left(\theta \mid B^a\right) = \max_{\|\theta' - \theta\| \le \rho} \mathcal{L}_{B^a}\left(\theta'\right) \approx \mathcal{L}_{B^a}\left(\theta + \eta_2 \nabla \mathcal{L}_{B^a}\left(\theta\right)\right)$ for a sufficient small learning rate $\eta_2 > 0$ (i.e., $\eta_2 \|\nabla \mathcal{L}_{B^a}\left(\theta\right)\| \le \rho$).

At each iteration, while primarily using the gradients of $\mathcal{L}\left(\theta \mid B^t\right)$ for optimization, we also utilize the gradient of $\mathcal{L}\left(\theta \mid B^a\right)$ to ensure congruent behavior between these two gradients. Specifically, at the l-th iteration, we update as follows:

$$\tilde{\theta}_l^a = \theta_l + \eta_2 \nabla_\theta \mathcal{L}_{B^a}\left(\theta_l\right), \tag{4}$$

$$\tilde{\theta}_l^t = \theta_l + \eta_1 \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) - \eta_2 \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right), \tag{5}$$

$$\theta_{l+1} = \theta_l - \eta \nabla_\theta \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right), \tag{6}$$

where $\eta_1 > 0, \eta_2 > 0$, and $\eta > 0$ are the learning rates, while $\mathcal{L}_{B^t}\left(\theta_l\right)$ and $\mathcal{L}_{B^a}\left(\theta_l\right)$ represent the empirical losses over the mini-batches $B^t, B^a \sim S^t$ respectively.

According to (6) (*i.e.*, $\theta_{l+1} = \theta_l - \eta \nabla_\theta \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)$), θ_{l+1} is updated to minimize $\mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)$. We now do first-order Taylor expansion for $\mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)$ as

$$\mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right) \approx \mathcal{L}_{B^t}\left(\theta_l\right) + \eta_1 \|\nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right)\|_2^2 - \eta_2 \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right), \tag{7}$$

where $\cdot$ specifies the dot product.

From (7), we reach the conclusion that the update in (6) aims to *minimize* simultaneously **(i)** $\mathcal{L}_{B^t}\left(\theta_l\right)$, **(ii)** $\|\nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right)\|_2^2$, and *maximize* **(iii)** $\nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)$. While the effects in (i) and (ii) are similar to SAM (Foret et al., 2021), maximizing $\nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)$) encourages two gradients of the losses over B^t and B^a to become more congruent. The following theorem shows that, during training, the two gradients of interest become *more congruent*.

Theorem 2. *For sufficiently small learning rates* $\eta_1 \le \frac{\left|\nabla_\theta \mathcal{L}_{B^t}(\theta_l) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)\right|}{12\left|\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^t}(\theta_l) \nabla_\theta \mathcal{L}_{B^t}(\theta_l)\right|}$ *and* $\eta_2 \le \min\left\{\frac{\left|\nabla_\theta \mathcal{L}_{B^t}(\theta_l) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)\right|}{6\left|\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^t}(\theta_l) \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)\right|}, \frac{\left|\nabla_\theta \mathcal{L}_{B^t}(\theta_l) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)\right|}{6\left|\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^a}\left(\tilde{\theta}_l^a\right) \nabla_\theta \mathcal{L}_{B^t}(\theta_l)\right|}\right\}$, *we have*

$$\nabla_\theta \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \ge \begin{cases} \frac{1}{2} \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \textit{if} \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \ge 0 \\ \frac{3}{2} \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \textit{otherwise} \end{cases} \tag{8}$$

Algorithm 1 Pseudo-code of Align-SAM

```
Input: $\rho, \lambda, \eta$, the number of iterations $T$, and the training set $S$
Output: the optimal model $\theta_T$.
for $l = 1$ to $T$ do
    Sample mini-batches $B^t, B^a \sim S$.
    $g_a = \nabla_\theta \mathcal{L}_{B^a}(\theta_l)$
    $\tilde{g}_a = \nabla_\theta \mathcal{L}_{B^a}\left(\theta_l + \rho \frac{g_a}{\|g_a\|_2}\right)$
    $g_t = \nabla_\theta \mathcal{L}_{B^t}(\theta_l)$
    Compute $\tilde{\theta}_l^t \leftarrow \theta_l + \rho\left(\lambda \frac{g_t}{\|g_t\|_2} - \frac{\tilde{g}_a}{\|\tilde{g}_a\|_2}\right)$.
    Compute $\theta_{l+1} \leftarrow \theta_l - \eta \nabla_\theta \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)$.
end for
```

Theorem 2 (see Appendix A.1 for proof) indicates that two gradients $\nabla_\theta \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)$ and $\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)$ are encouraged to be more congruent since our update aims to maximize its lower bound $c \times \nabla_\theta \mathcal{L}_{B^t}(\theta_l) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)$ (i.e., $c = 0.5$ or $c = 1.5$).

Practical Algorithm. Inspired by SAM (Foret et al., 2021), we set $\eta_2 = \frac{\rho}{\|\nabla_\theta \mathcal{L}_{B^a}(\theta_l)\|_2}$ and $\eta_1 = \lambda \frac{\rho}{\|\nabla_\theta \mathcal{L}_{B^t}(\theta_l)\|_2}$, where $\rho > 0$ are perturbation radius and λ is trade-off coefficient for combining gradient from B^t and B^a. In practice, we observe that setting $\lambda > 1$, which prioritizes the gradient from the training mini-batch B^t, results in improved performance. This trade-off is discussed in Section A.3.

The pseudo-code of Align-SAM is presented in Algorithm 1. Compared to standard SAM, our method requires additional forward and backward passes due to the use of an auxiliary batch. To reduce this overhead, we set the auxiliary batch size $|B^a|$ significantly smaller than the primary training batch size $|B^t|$, ensuring that most computation is devoted to the main update step. As a result, Align-SAM is only marginally slower than standard SAM, as reported in Table 9. Further details are provided in the Appendix.

3.3 Convergence Analysis

It is well known that the normalized (practical) version of SAM *does not converge* to the minimizer of the training loss, as rigorously demonstrated in Si & Yun (2023) (Theorem 4.6), one of the most comprehensive analyses of SAM's convergence behavior. Our proposed approach shares the *same convergence rate* as standard SAM, as established in Si & Yun (2023) (Theorem 4.6). Details in Appendix A.2

4 Experiments

In this section, we present the results of various experiments to evaluate the effectiveness of our Align-SAM, including training from scratch, transfer learning on different dataset sizes, learning with noisy labels, and a meta-learning setting.

4.1 Image Classification From Scratch

We first conduct experiments on ImageNet, Food101, and CIFAR datasets with standard image classification settings trained from scratch. The performance is compared with baseline models trained with the SGD, SAM, ASAM, and the integration of ASAM and Align-SAM. For all experiments of Align-SAM, we consistently set $\lambda = 2$ and discuss the effect of this trade-off in Section A.3.

ImageNet dataset. We use ResNet18 and ResNet34 models for experiments on the ImageNet dataset, with an input size of 224×224. For all experiments with Align-SAM, we consistently set $\lambda = 2$, while the perturbation radius ρ is configured according to the SAM method. Specifically, in this experiment, we set $\rho = 0.1$ for both SAM and Align-SAM. The models are trained for 200 epochs with basic data augmentations (random cropping, horizontal flipping, and normalization). We use an initial learning rate of 0.1, a batch size of 2048 for the training mini-batches, and 512 for the auxiliary mini-batches, following a cosine learning schedule across all experiments in this paper. We extend this experiment to the mid-sized Food101 dataset using the same settings, except for a batch size of 128 for the training and 32 for the auxiliary mini-batches. Performance results are detailed in Table 1.

Table 1: Classification accuracy on the ImageNet and Food101 datasets. All models are trained from scratch with 200 epochs.

Dataset	Method	Resnet18		Resnet34	
		Top-1	Top-5	Top-1	Top-5
ImageNet	SAM	62.46	84.19	63.73	84.95
	Align-SAM	**63.64**	**85.22**	**65.89**	**86.84**
Food101	SAM	73.15	89.85	73.87	90.84
	Align-SAM	**73.45**	**90.35**	**74.47**	**91.27**

CIFAR dataset. We used three architectures: WideResnet28x10, Pyramid101, and Densenet121 with an input size of 32×32 for CIFAR datasets. To replicate the baseline experiments, we followed the hyperparameters provided in the original papers. Specifically, for CIFAR-100, we set $\rho = 0.1$, and for CIFAR-10, we used $\rho = 0.05$ for SAM, VASSO and Align-SAM. The same procedure and settings were applied to ASAM and Align-ASAM, with the perturbation radius $\rho = 1.0$ for CIFAR-100 and $\rho = 0.5$ for CIFAR-10. Other training configurations are consistent with those used in the ImageNet experiments, except for data augmentations (horizontal flipping, four-pixel padding, and random cropping). We use $\theta = 0.9$ as the default parameter for VASSO. The results are reproduced and reported in Tables 2, while the SGD results are referenced from Foret et al. (2021).

Our proposed method outperforms the baselines across various settings. On both ImageNet and Food101, it significantly surpasses the baselines, with a notable improvement in both Top-1 and Top-5 accuracy. For CIFAR-10, performance is close to the saturation point, making further improvements challenging. Nevertheless, Align-SAM achieves slight enhancements across all cases. On CIFAR-100, where models are more prone to overfitting compared to CIFAR-10, Align-SAM still delivers competitive results.

4.2 Transfer learning

In this subsection, we further evaluate Align-SAM in the transfer learning setting using the ImageNet pre-trained models to fine-tune both small-size, mid-size, and large-size datasets. All initialized weights are available on the Pytorch library.

First, we conduct experiments on ImageNet using three models from the ResNet family and a ViT-Adapter-S (which incorporates lightweight Adapter modules with a plain ViT-Small backbone). The ResNet models are pre-trained on ImageNet, while the backbone ViT-Small of ViT-Adapter-S is pre-trained on ImageNet-21k. Each model is then fine-tuned for 50 epochs using either SAM or Align-SAM with a learning rate of 0.01. We set $\rho = 0.05$ for SAM and Align-SAM; basic augmentation techniques are the same as training from the scratch setting. Results reported in Table 3 show that our methods outperform baselines with a significant gap in both top-1 and top-5 accuracies.

Next, we examine this setting on small and mid-sized datasets on three models of the EfficientNet family. We fine-tune with a learning rate of 0.05 in 50 epochs and use $\rho = 0.1$ for all experiments of SAM, VASSO (with $\theta = 0.9$ as the default) Li & Giannakis (2024), and Align-SAM. In Table 4, Align-SAM achieves a noticeable improvement compared to most of the baselines on all small-size, mid-size, and large-size datasets, demonstrating its robustness and stability across various experiment settings.

Table 2: Classification accuracy on the CIFAR datasets. All models are trained from scratch three times with different random seeds and we report the mean and standard deviation of accuracies.

Method	WideResnet28x10	Pyramid101	Densenet121
Dataset CIFAR-100			
SGD Foret et al. (2021)	81.20 ± 0.200	80.30 ± 0.300	-
SAM Foret et al. (2021)	83.00 ± 0.035	81.99 ± 0.636	68.72 ± 0.409
VASSO Li & Giannakis (2024)	83.11 ± 0.063	82.04 ± 0.127	69.00 ± 0.261
GSAM Zhuang et al. (2022)	83.13 ± 0.099	81.87 ± 0.143	68.88 ± 0.201
LookSAM (k=1) Liu et al. (2022)	82.89 ± 0.111	82.25 ± 0.273	69.05 ± 0.182
Align-SAM	**83.72 ± 0.049**	**82.53 ± 0.282**	**69.10 ± 0.311**
ASAM	83.16 ± 0.296	82.02 ± 0.134	69.62 ± 0.120
Align-ASAM	**83.88 ± 0.042**	**82.31 ± 0.183**	**69.71 ± 0.339**
Dataset CIFAR-10			
SGD Foret et al. (2021)	96.50 ± 0.100	96.00 ± 0.100	-
SAM Foret et al. (2021)	96.87 ± 0.027	96.17 ± 0.174	91.28 ± 0.241
VASSO Li & Giannakis (2024)	96.84 ± 0.014	96.22 ± 0.035	91.18 ± 0.063
GSAM Zhuang et al. (2022)	96.91 ± 0.020	96.15 ± 0.113	91.50 ± 0.109
LookSAM(k=1) Liu et al. (2022)	**97.00 ± 0.181**	96.00 ± 0.207	91.10 ± 0.196
Align-SAM	96.91 ± 0.007	**96.47 ± 0.219**	**91.54 ± 0.307**
ASAM Kwon et al. (2021)	96.91 ± 0.063	96.45 ± 0.042	**92.04 ± 0.240**
Align-ASAM	**97.15 ± 0.063**	**96.56 ± 0.261**	92.02 ± 0.000

Table 3: Transfer learning on ImageNet with Resnet models.

Model	Top-1 Acc		Top-5 Acc	
	SAM Foret et al. (2021)	Align-SAM	SAM Foret et al. (2021)	Align-SAM
Resnet18	70.44 ± 0.12	**70.92 ± 0.05**	89.63 ± 0.04	**89.90 ± 0.04**
Resnet34	73.06 ± 0.48	**73.94 ± 0.14**	91.29 ± 0.03	**91.81 ± 0.03**
Resnet50	75.17 ± —	**75.91 ± —**	92.58 ± —	**92.83 ± —**
ViT-Adapter-S	79.96 ± —	**80.04 ± —**	95.35 ± —	**95.39 ± —**

Table 4: Transfer learning accuracy of small and medium datasets. All models are fine-tuned from pre-trained weights on ImageNet.

Dataset	Top-1 Acc				Top-5 Acc			
	SGD	SAM	VASSO	Align-SAM	SGD	SAM	VASSO	Align-SAM
EfficientNet-B2								
Stanford Cars	89.14 ± 0.11	89.68 ± 0.17	89.91 ± 0.24	**90.39 ± 0.07**	97.60 ± 0.20	98.04 ± 0.07	98.03 ± 0.09	**98.30 ± 0.09**
FGVC-Aircraft	85.83 ± 0.23	86.25 ± 0.36	86.23 ± 0.22	**87.22 ± 0.27**	95.72 ± 0.02	95.87 ± 0.06	95.94 ± 0.05	**95.94 ± 0.03**
Oxford IIIT Pets	92.17 ± 0.19	92.34 ± 0.11	92.40 ± 0.16	**92.64 ± 0.17**	99.23 ± 0.02	99.35 ± 0.02	99.34 ± 0.01	**99.35 ± 0.07**
Flower102	95.06 ± 0.01	95.22 ± 0.14	95.37 ± 0.11	**95.43 ± 0.10**	99.08 ± 0.18	99.11 ± 0.19	**99.23 ± 0.09**	99.18 ± 0.02
Food101	83.50 ± 0.01	85.12 ± 0.07	84.65 ± 0.03	**85.51 ± 0.02**	96.10 ± 0.32	96.83 ± 0.08	96.60 ± 0.09	**97.14 ± 0.00**
Country211	11.94 ± 0.14	12.48 ± 0.03	12.49 ± 0.10	**13.41 ± 0.00**	23.70 ± 0.13	25.49 ± 0.07	24.90 ± 0.13	**27.06 ± 0.16**
EfficientNet-B3								
Stanford Cars	89.01 ± 0.19	89.40 ± 0.09	89.55 ± 0.12	**89.86 ± 0.14**	97.73 ± 0.21	98.03 ± 0.07	98.01 ± 0.05	**98.10 ± 0.01**
FGVC-Aircraft	84.88 ± 0.08	85.19 ± 0.11	85.15 ± 0.19	**85.78 ± 0.25**	95.53 ± 0.12	95.67 ± 0.00	95.67 ± 0.02	**96.26 ± 0.10**
Oxford IIIT Pets	92.68 ± 0.25	92.58 ± 0.02	92.64 ± 0.06	**92.75 ± 0.19**	99.00 ± 0.01	99.19 ± 0.05	99.23 ± 0.10	**99.26 ± 0.11**
Flower102	94.59 ± 0.10	94.73 ± 0.14	94.94 ± 0.17	**95.32 ± 0.26**	98.95 ± 0.08	99.12 ± 0.16	99.13 ± 0.07	**99.26 ± 0.07**
Food101	83.75 ± 0.12	85.79 ± 0.13	85.69 ± 0.14	**85.95 ± 0.13**	96.22 ± 0.02	97.12 ± 0.00	97.07 ± 0.09	**97.33 ± 0.00**
Country211	12.96 ± 0.01	13.38 ± 0.09	13.36 ± 0.08	**13.61 ± 0.05**	26.11 ± 0.56	25.78 ± 0.08	25.91 ± 0.05	**26.71 ± 0.26**
EfficientNet-B4								
Stanford Cars	84.72 ± 0.04	85.08 ± 0.16	85.06 ± 0.07	**85.46 ± 0.32**	96.41 ± 0.07	96.45 ± 0.01	96.53 ± 0.04	**96.81 ± 0.00**
FGVC-Aircraft	79.95 ± 0.61	79.96 ± 0.04	80.02 ± 0.38	**80.53 ± 0.51**	94.87 ± 0.08	94.65 ± 0.08	94.68 ± 0.13	**94.74 ± 0.01**
Oxford IIIT Pets	91.89 ± 0.13	92.02 ± 0.23	92.04 ± 0.18	**92.07 ± 0.00**	99.28 ± 0.10	99.43 ± 0.07	**99.59 ± 0.08**	99.44 ± 0.02
Flower102	92.73 ± 0.04	93.02 ± 0.14	93.02 ± 0.16	**93.07 ± 0.16**	98.49 ± 0.07	98.68 ± 0.02	**98.73 ± 0.07**	98.63 ± 0.05
Food101	84.55 ± 0.14	86.13 ± 0.06	86.18 ± 0.10	**86.40 ± 0.44**	96.31 ± 0.03	97.07 ± 0.01	97.07 ± 0.03	**97.31 ± 0.02**
Country211	14.63 ± 0.09	14.80 ± 0.13	14.97 ± 0.11	**15.26 ± 0.16**	27.60 ± 0.00	28.09 ± 1.77	28.00 ± 0.18	**28.24 ± 0.14**

4.3 Train With Noisy Label

In addition to mitigating data shifts between training and testing datasets, we evaluate the robustness of Align-SAM against noisy labels on the standard training procedure. Specifically, we adopt a classical noisy-label setting for CIFAR-10 and CIFAR-100, in which a portion of the training set's labels are symmetrically flipped with noise fractions {0.2, 0.4, 0.6, 0.8}, while the testing set's labels remain unchanged.

Table 5: Results under label noise on CIFAR datasets with ResNet32. Each experiment is conducted three times using different random seeds, and we report their averages and standard deviations.

Method	Noise rate (%)			
	0.2	0.4	0.6	0.8
Dataset CIFAR-100				
SGD	66.22 ± 0.355	59.26 ± 0.045	46.77 ± 0.020	26.49 ± 0.640
SAM	66.16 ± 0.721	59.95 ± 0.622	50.81 ± 0.353	24.26 ± 1.209
FSAM	65.73 ± 0.219	58.96 ± 0.381	49.36 ± 1.103	25.92 ± 1.173
VASSO ($\theta = 0.9$)	66.52 ± 0.254	59.67 ± 0.318	50.09 ± 0.353	20.85 ± 0.077
VASSO ($\theta = 0.2$)	65.16 ± 0.042	59.07 ± 0.820	48.35 ± 1.046	**28.49 ± 0.551**
Align-SAM	**66.78 ± 0.657**	**60.78 ± 0.636**	**51.03 ± 0.502**	27.66 ± 1.265
ASAM	66.88 ± 0.593	61.53 ± 0.487	52.77 ± 0.561	30.33 ± 1.788
Align-ASAM	**67.38 ± 0.106**	**62.72 ± 0.304**	**54.58 ± 0.572**	**32.77 ± 0.388**
Dataset CIFAR-10				
SGD	89.98 ± 0.070	84.83 ± 0.085	75.06 ± 0.385	54.47 ± 1.265
SAM	91.26 ± 0.007	88.19 ± 1.060	83.43 ± 0.622	61.69 ± 0.289
FSAM	91.35 ± 0.318	87.58 ± 0.353	82.78 ± 2.057	58.09 ± 2.276
VASSO ($\theta = 0.9$)	91.47 ± 0.487	88.17 ± 0.890	83.75 ± 0.480	67.71 ± 4.129
VASSO ($\theta = 0.2$)	90.45 ± 0.855	86.28 ± 0.997	77.33 ± 0.806	**70.95 ± 0.770**
Align-SAM	**92.38 ± 0.007**	**90.20 ± 0.318**	**85.33 ± 0.268**	70.02 ± 0.403
ASAM	91.98 ± 0.007	89.24 ± 0.572	84.39 ± 0.445	64.82 ± 6.880
Align-ASAM	**92.06 ± 0.367**	**90.01 ± 0.282**	**86.09 ± 0.657**	**73.25 ± 0.353**

All experiments are conducted using the ResNet32 architecture, with models trained from scratch for 200 epochs. The batch size is set to 512 for the training mini-batches and 128 for the auxiliary mini-batches. Following Foret et al. (2021), we set $\rho = 0.1$ SAM, FSAM Li et al. (2024), VASSO Li & Giannakis (2024), and Align-SAM, $\rho = 1.0$ for ASAM and Align-ASAM when training with all noise levels. Exceptionally, for the setting with 80% noisy labels, the perturbation radius for SAM, FSAM, Align-SAM, ASAM, and Align-ASAM is reduced by half to ensure more stable convergence. This observation is also noted in Li et al. (2024) and Foret et al. (2021).

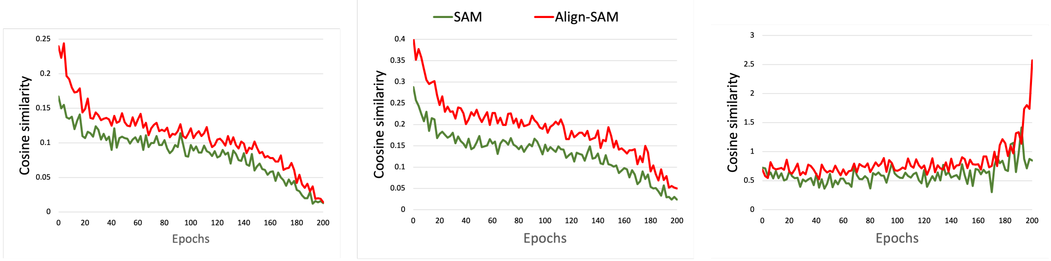
In line with Li et al. (2024), we apply additional cutout techniques along with the basic augmentations outlined in Section 4.1. We report the results of VASSO with $\theta = \{0.2, 0.9\}$, as presented in their paper Li & Giannakis (2024), where $\theta = 0.2$ is expected to yield better performance. However, we observe that $\theta = 0.2$ performs better only in the setting with 80% noisy labels, while in other noisy label settings, it gives a lower result compared to $\theta = 0.9$. Each experiment is repeated three times with different random seeds, and we report the average and standard deviation of the results in Table 5. Note that training with SGD is prone to overfitting as the number of epochs increases. Therefore, we present the best results for SGD training at both 200 and 400 epochs.

5 Ablation study

5.1 Cosine similarity of gradients

We present the cosine similarity of gradients before and update the model using SAM and Align-SAM in Figure 1. Detail analysis is presented in Appendix A.3.

The size of the auxiliary subset $|B^a|$ and complexity, the sensitivity of trade-off coefficient λ, the analysis of loss landscape, and details of these experiments are presented in Appendix A.3.



(a) Before updating model

(b) After updating model

(c) Improvement after updating

Figure 1: Cosine similarity of two gradients $\nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right)$ and $\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)$ (**a**) before updating model, (**b**) after updating model and (**c**) the improvement of this similarity.

6 Conclusion and Limitation

In conclusion, this work revisits Sharpness-Aware Minimization (SAM) through the lens of cross-subset alignment, offering a fresh perspective on generalization. While SAM encourages flat minima to improve generalization, we argue that effective generalization also hinges on the alignment between two independently sampled subsets from the same data distribution. Building on this insight, we introduce *Align-SAM* to find models that minimize sharpness and loss on the primary subset while simultaneously maintaining low loss on the auxiliary subset, thereby ensuring stability across resamplings of the data distribution. By explicitly aligning the optimization process across both subsets, Align-SAM produces models that are not only robust to perturbations but also more resilient to distributional shifts. Extensive experiments confirm that Align-SAM delivers consistent gains in generalization, particularly under challenging conditions such as label noise and data scarcity. One limitation to note is that using an additional auxiliary subset in each training iteration may increase training time (depending on the size of the auxiliary sets). We view this as a trade-off between performance and training complexity. However, this issue could potentially be mitigated by reusing the gradients from the previous steps. We leave this as a direction for future work to reduce training complexity while maintaining performance.

References

Momin Abbas, Quan Xiao, Lisha Chen, Pin-Yu Chen, and Tianyi Chen. Sharp-maml: Sharpness-aware model-agnostic meta learning. *arXiv preprint arXiv:2206.03996*, 2022.

Pierre Alquier. User-friendly introduction to pac-bayes bounds, 2023. URL `https://arxiv.org/abs/2110.11216`.

Pierre Alquier, James Ridgway, and Nicolas Chopin. On the properties of variational approximations of gibbs posteriors. *Journal of Machine Learning Research*, 17(236), 2016a. URL `http://jmlr.org/papers/v17/15-290.html`.

Pierre Alquier, James Ridgway, and Nicolas Chopin. On the properties of variational approximations of gibbs posteriors. *The Journal of Machine Learning Research*, 17(1):8374–8414, 2016b.

Maksym Andriushchenko and Nicolas Flammarion. Towards understanding sharpness-aware minimization. In *International Conference on Machine Learning*, pp. 639–668. PMLR, 2022a.

Maksym Andriushchenko and Nicolas Flammarion. Towards understanding sharpness-aware minimization. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 639–668. PMLR, 17–23 Jul 2022b. URL `https://proceedings.mlr.press/v162/andriushchenko22a.html`.

Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International conference on machine learning*, pp. 233–242. PMLR, 2017.

Dara Bahri, Hossein Mobahi, and Yi Tay. Sharpness-aware minimization improves language model generalization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7360–7371, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.508. URL `https://aclanthology.org/2022.acl-long.508`.

Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *J. Mach. Learn. Res.*, 3(null):463–482, mar 2003. ISSN 1532-4435.

Olivier Bousquet and André Elisseeff. Stability and generalization. *The Journal of Machine Learning Research*, 2:499–526, 2002.

Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. *Advances in Neural Information Processing Systems*, 34:22405–22418, 2021.

Xiangning Chen, Cho-Jui Hsieh, and Boqing Gong. When vision transformers outperform resnets without pre-training or strong data augmentations. *arXiv preprint arXiv:2106.01548*, 2021.

Zixiang Chen, Junkai Zhang, Yiwen Kou, Xiangning Chen, Cho-Jui Hsieh, and Quanquan Gu. Why does sharpness-aware minimization generalize better than sgd? In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Jiaxin Deng, Junbiao Pang, Baochang Zhang, and Guodong Guo. Asymptotic unbiased sample sampling to speed up sharpness-aware minimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 16208–16216, 2025.

Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In *International Conference on Machine Learning*, pp. 1019–1028. PMLR, 2017.

Jiawei Du, Daquan Zhou, Jiashi Feng, Vincent Tan, and Joey Tianyi Zhou. Sharpness-aware training for free. *Advances in Neural Information Processing Systems*, 35:23439–23451, 2022.

Gintare Karolina Dziugaite and Daniel M. Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. In *UAI*. AUAI Press, 2017a.

Gintare Karolina Dziugaite and Daniel M Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *arXiv preprint arXiv:1703.11008*, 2017b.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1126–1135. PMLR, 06–11 Aug 2017.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=6Tm1mposlrM`.

Stanislav Fort and Surya Ganguli. Emergent properties of the local geometry of neural loss landscapes. *arXiv preprint arXiv:1910.05929*, 2019.

Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1568–1577. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/franceschi18a.html`.

Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory F. Diamos, Heewoo Jun, Hassan Kianinejad, Md. Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *CoRR*, abs/1712.00409, 2017.

Sepp Hochreiter and Jürgen Schmidhuber. Simplifying neural nets by discovering flat minima. In *NIPS*, pp. 529–536. MIT Press, 1994.

Bin Huang, Ying Xie, and Chaoyang Xu. Learning with noisy labels via clean aware sharpness aware minimization. *Scientific Reports*, 15(1):1350, 2025.

Stanislaw Jastrzebski, Zachary Kenton, Devansh Arpit, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos J. Storkey. Three factors influencing minima in sgd. *ArXiv*, abs/1711.04623, 2017.

Jie Ji, Gen Li, Jingjing Fu, Fatemeh Afghah, Linke Guo, Xiaoyong Yuan, and Xiaolong Ma. A single-step, sharpness-aware minimization is all you need to achieve efficient and accurate sparse training. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 44269–44290. Curran Associates, Inc., 2024. URL `https://proceedings.neurips.cc/paper_files/paper/2024/file/4e5a47c427ec16ec863e547bc1aeb70c-Paper-Conference.pdf`.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *ICLR*. OpenReview.net, 2020.

Jean Kaddour, Linqing Liu, Ricardo Silva, and Matt J Kusner. A fair comparison of two popular flat minima optimizers: Stochastic weight averaging vs. sharpness-aware minimization. *arXiv preprint arXiv:2202.00661*, 1, 2022.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *CoRR*, abs/2001.08361, 2020.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *ICLR*. OpenReview.net, 2017.

Minyoung Kim, Da Li, Shell X Hu, and Timothy Hospedales. Fisher SAM: Information geometry and sharpness aware minimisation. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 11148–11161. PMLR, 17–23 Jul 2022.

Jungmin Kwon, Jeongseop Kim, Hyunseo Park, and In Kwon Choi. Asam: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. In *International Conference on Machine Learning*, pp. 5905–5914. PMLR, 2021.

Bingcong Li and Georgios Giannakis. Enhancing sharpness-aware optimization through variance suppression. *Advances in Neural Information Processing Systems*, 36, 2024.

Sicong Li, Qianqian Xu, Zhiyong Yang, Zitai Wang, Linchao Zhang, Xiaochun Cao, and Qingming Huang. Focal-SAM: Focal sharpness-aware minimization for long-tailed classification. In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=lCk4PZto8T`.

Tao Li, Pan Zhou, Zhengbao He, Xinwen Cheng, and Xiaolin Huang. Friendly sharpness-aware minimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5631–5640, 2024.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

Yong Liu, Siqi Mai, Xiangning Chen, Cho-Jui Hsieh, and Yang You. Towards efficient and scalable sharpness-aware minimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12360–12370, 2022.

Haocheng Luo, Tuan Truong, Tung Pham, Mehrtash Harandi, Dinh Phung, and Trung Le. Explicit eigenvalue regularization improves sharpness-aware minimization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL `https://openreview.net/forum?id=JFUhBY34SC`.

David A. McAllester. Pac-bayesian model averaging. In *Proceedings of the Twelfth Annual Conference on Computational Learning Theory*, COLT '99, pp. 164–170, New York, NY, USA, 1999. Association for Computing Machinery. ISBN 1581131674. doi: 10.1145/307400.307435. URL `https://doi.org/10.1145/307400.307435`.

Thomas Möllenhoff and Mohammad Emtiyaz Khan. Sam as an optimal relaxation of bayes. *arXiv preprint arXiv:2210.01620*, 2022.

Enea Monzio Compagnoni, Luca Biggio, Antonio Orvieto, Frank Norbert Proske, Hans Kersting, and Aurelien Lucchi. An SDE for modeling SAM: Theory and insights. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 25209–25253. PMLR, 23–29 Jul 2023. URL `https://proceedings.mlr.press/v202/monzio-compagnoni23a.html`.

Gonçalo Mordido, Pranshu Malviya, Aristide Baratin, and Sarath Chandar. Lookbehind-SAM: k steps back, 1 step forward. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=vCN5lwcWWE`.

Sayan Mukherjee, Partha Niyogi, Tomaso A. Poggio, and Ryan M. Rifkin. Statistical learning: Stability is sufficient for generalization and necessary and sufficient for consistency of empirical risk minimization. 2002. URL `https://api.semanticscholar.org/CorpusID:7478036`.

Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nati Srebro. Exploring generalization in deep learning. *Advances in neural information processing systems*, 30, 2017.

Tuan H Nguyen, Paymon Haddad, Eric Gan, and Baharan Mirzasoleiman. Changing the training data distribution to reduce simplicity bias improves in-distribution generalization. *Advances in Neural Information Processing Systems*, 37:68854–68896, 2024.

Van-Anh Nguyen, Tung-Long Vuong, Hoang Phan, Thanh-Toan Do, Dinh Phung, and Trung Le. Flat seeking bayesian neural network. In *Advances in Neural Information Processing Systems*, 2023.

Henning Petzka, Michael Kamp, Linara Adilova, Cristian Sminchisescu, and Mario Boley. Relative flatness and generalization. In *NeurIPS*, pp. 18420–18432, 2021.

Hoang Phan, Ngoc Tran, Trung Le, Toan Tran, Nhat Ho, and Dinh Phung. Stochastic multiple target sampling gradient descent. *Advances in neural information processing systems*, 2022.

Tomaso Poggio, Ryan Rifkin, Sayan Mukherjee, and Partha Niyogi. General conditions for predictivity in learning theory. *Nature*, 428(6981):419–422, 2004.

Zhe Qu, Xingyu Li, Rui Duan, Yao Liu, Bo Tang, and Zhuo Lu. Generalized federated learning via sharpness aware minimization. *arXiv preprint arXiv:2206.02618*, 2022.

Dongkuk Si and Chulhee Yun. Practical sharpness-aware minimization cannot converge all the way to optima. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

Chengli Tan, Jiangshe Zhang, Junmin Liu, Yicheng Wang, and Yunda Hao. Stabilizing sharpness-aware minimization through a simple renormalization strategy. *Journal of Machine Learning Research*, 26(68):1–35, 2025.

Vladimir Naumovich Vapnik. Statistical learning theory. In *Adaptive and Learning Systems for Signal Processing, Communications, and Control*. Wiley, 1998.

Colin Wei, Sham Kakade, and Tengyu Ma. The implicit and explicit regularization effects of dropout. In *International conference on machine learning*, pp. 10181–10192. PMLR, 2020.

Kaiyue Wen, Zhiyuan Li, and Tengyu Ma. Sharpness minimization algorithms do not only minimize sharpness to achieve better generalization. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a. URL `https://openreview.net/forum?id=Dkmpa6wCIx`.

Kaiyue Wen, Tengyu Ma, and Zhiyuan Li. How sharpness-aware minimization minimizes sharpness? In *The Eleventh International Conference on Learning Representations*, 2023b. URL `https://openreview.net/forum?id=5spDgWmpY6x`.

Xinda Xing, Qiugang Zhan, Xiurui Xie, Yuning Yang, Qiang Wang, and Guisong Liu. Flexible sharpness-aware personalized federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 21707–21715, 2025.

Runsheng Yu, Youzhi Zhang, and James Kwok. Improving sharpness-aware minimization by lookahead. In *Forty-first International Conference on Machine Learning*, 2024.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Xingxuan Zhang, Renzhe Xu, Han Yu, Hao Zou, and Peng Cui. Gradient norm aware minimization seeks first-order flatness and improves generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20247–20257, 2023.

Yixuan Zhou, Yi Qu, Xing Xu, and Hengtao Shen. Imbsam: A closer look at sharpness-aware minimization in class-imbalanced recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11345–11355, 2023a.

Zhipeng Zhou, Lanqing Li, Peilin Zhao, Pheng-Ann Heng, and Wei Gong. Class-conditional sharpness-aware minimization for deep long-tailed recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3499–3509, 2023b.

Juntang Zhuang, Boqing Gong, Liangzhe Yuan, Yin Cui, Hartwig Adam, Nicha C. Dvornek, Sekhar Tatikonda, James S. Duncan, and Ting Liu. Surrogate gap minimization improves sharpness aware training. In *International Conference on Learning Representations (ICLR)*, 2022. URL `https://arxiv.org/abs/2203.08065`.

A Appendix / supplemental material

In this appendix, we present the proofs in our paper and additional experiments. We open-source our code and provide instruction, scripts, and log files to reproduce experiments at `https://anonymous.4open.science/r/AlignSAM-43CD/README.md`

A.1 All Proofs

Proof of Theorem 1

Proof. We use the PAC-Bayes theory in this proof. In PAC-Bayes theory, θ could follow a distribution, says P, thus we define the expected loss over θ distributed by P as follows:

$$\mathcal{L}_{\mathcal{D}}(\theta, P) = \mathbb{E}_{\theta\sim P}\big[\mathcal{L}_{\mathcal{D}}(\theta)\big]$$
$$\mathcal{L}_{\mathcal{S}}(\theta, P) = \mathbb{E}_{\theta\sim P}\big[\mathcal{L}_{\mathcal{S}}(\theta)\big].$$

For any distribution $P = \mathcal{N}(\mathbf{0}, \sigma_P^2 \mathbb{I}_k)$ and $Q = \mathcal{N}(\theta, \sigma^2 \mathbb{I}_k)$ over $\theta \in \mathbb{R}^k$, where P is the prior distribution and Q is the posterior distribution, use the PAC-Bayes theorem in Alquier et al. (2016a), for all $\beta > 0$, with a probability at least $1 - \delta$, we have

$$\mathcal{L}_{\mathcal{D}}(\theta, Q) \le \mathcal{L}_{\mathcal{S}}(\theta, Q) + \frac{1}{\beta}\Big[\mathsf{KL}(Q\|P) + \log\frac{1}{\delta} + \Psi(\beta, N)\Big], \tag{9}$$

where Ψ is defined as

$$\Psi(\beta, N) = \log \mathbb{E}_P \mathbb{E}_{\mathcal{D}^N}\Big[\exp\big\{\beta\big[\mathcal{L}_{\mathcal{D}}(f_\theta) - \mathcal{L}_{\mathcal{S}}(f_\theta)\big]\big\}\Big].$$

When the loss function is bounded by L, then

$$\Psi(\beta, N) \le \frac{\beta^2 L^2}{8N}.$$

The task is to minimize the second term of RHS of (9), we thus choose $\beta = \sqrt{8N}\frac{\mathsf{KL}(Q\|P)+\log\frac{1}{\delta}}{L}$. Then the second term of RHS of (9) is equal to

$$\sqrt{\frac{\mathsf{KL}(Q\|P) + \log\frac{1}{\delta}}{2N}} \times L.$$

The KL divergence between Q and P, when they are Gaussian, is given by formula

$$\mathsf{KL}(Q\|P) = \frac{1}{2}\left[\frac{k\sigma^2 + \|\theta\|^2}{\sigma_P^2} - k + k\log\frac{\sigma_P^2}{\sigma^2}\right].$$

For given posterior distribution Q with fixed σ^2, to minimize the KL term, the σ_P^2 should be equal to $\sigma^2 + \|\theta\|^2/k$. In this case, the KL term is no less than

$$k\log\Big(1 + \frac{\|\theta_0\|^2}{k\sigma^2}\Big).$$

Thus, the second term of RHS is

$$\sqrt{\frac{\mathsf{KL}(Q\|P) + \log\frac{1}{\delta}}{2N}} \times L \ge \sqrt{\frac{k\log\big(1 + \frac{\|\theta\|^2}{k\sigma^2}\big)}{4N}} \times L \ge L$$

when $\|\theta\|^2 > \sigma^2\big\{\exp(4N/k) - 1\big\}$. Hence, for any $\|\theta\|_2 > \sigma^2\big\{\exp(4N/k) - 1\big\}$, we have the RHS is greater than the LHS, and the inequality is trivial. In this work, we only consider the case:

$$\|\theta\|^2 < \sigma^2\big(\exp\{4N/k\} - 1\big). \tag{10}$$

Distribution P is Gaussian centered around $\mathbf{0}$ with variance $\sigma_P^2 = \sigma^2 + \|\theta\|^2/k$, which is unknown at the time we set up the inequality, since θ is unknown. Meanwhile, we have to specify P in advance, since P is the prior distribution. To deal with this problem, we could choose a family of P such that its means cover the space of θ satisfying inequality (10). We set

$$\begin{aligned} c &= \sigma^2\big(1+\exp\{4N/k\}\big) \\ P_j &= \mathcal{N}\big(0, c\exp\frac{1-j}{k}\mathbb{I}_k\big) \\ \mathfrak{P} &:= \big\{P_j : j = 1,2,\dots\big\} \end{aligned}$$

Then the following inequality holds for a particular distribution P_j with probability $1-\delta_j$ with $\delta_j = \frac{6\delta}{\pi^2 j^2}$

$$\mathbb{E}_{\theta'\sim\mathcal{N}(\theta,\sigma^2)}\mathcal{L}_{\mathcal{D}}\big(f_{\theta'}\big) \le \mathbb{E}_{\theta'\sim\mathcal{N}(\theta,\sigma^2)}\mathcal{L}_{\mathcal{S}}\big(f_{\theta'}\big) + \frac{1}{\beta}\left[\mathsf{KL}(Q\|P_j) + \log\frac{1}{\delta_j} + \Psi(\beta,N)\right].$$

Use the well-known equation: $\sum_{j=1}^{\infty}\frac{1}{j^2} = \frac{\pi^2}{6}$, then with probability $1-\delta$, the above inequality holds with every j. We pick

$$j^* := \left\lfloor 1 - k\log\frac{\sigma^2+\|\theta\|^2/k}{c}\right\rfloor = \left\lfloor 1 - k\log\frac{\sigma^2+\|\theta\|^2/k}{\sigma^2(1+\exp\{4N/k\})}\right\rfloor.$$

Therefore,

$$\begin{aligned} 1-j^* &= \left\lceil k\log\frac{\sigma^2+\|\theta\|^2/k}{c}\right\rceil \\ \Rightarrow\quad \log\frac{\sigma^2+\|\theta\|^2/k}{c} &\le \frac{1-j^*}{k} \le \log\frac{\sigma^2+\|\theta_0\|^2/k}{c} + \frac{1}{k} \\ \Rightarrow\quad \sigma^2+\|\theta\|^2/k &\le c\exp\left\{\frac{1-j^*}{k}\right\} \le \exp(1/k)\big[\sigma^2+\|\theta\|^2/k\big] \\ \Rightarrow\quad \sigma^2+\|\theta\|^2/k &\le \sigma^2_{P_{j^*}} \le \exp(1/k)\big[\sigma^2+\|\theta\|^2/k\big]. \end{aligned}$$

Thus the KL term could be bounded as follow

$$\begin{aligned} \mathsf{KL}(Q\|P_{j^*}) &= \frac{1}{2}\left[\frac{k\sigma^2+\|\theta\|^2}{\sigma^2_{P_{j^*}}} - k + k\log\frac{\sigma^2_{P_{j^*}}}{\sigma^2}\right] \\ &\le \frac{1}{2}\left[\frac{k(\sigma^2+\|\theta\|^2/k)}{\sigma^2+\|\theta\|^2/k} - k + k\log\frac{\exp(1/k)\big(\sigma^2+\|\theta\|^2/k\big)}{\sigma^2}\right] \\ &= \frac{1}{2}\left[k\log\frac{\exp(1/k)\big(\sigma^2+\|\theta\|^2/k\big)}{\sigma^2}\right] \\ &= \frac{1}{2}\left[1 + k\log\Big(1+\frac{\|\theta_0\|^2}{k\sigma^2}\Big)\right] \end{aligned}$$

For the term $\log\frac{1}{\delta_{j^*}}$, with recall that $c = \sigma^2\big(1+\exp(4N/k)\big)$ and

$j^* = \left\lfloor 1 - k\log\frac{\sigma^2+\|\theta\|^2/k}{\sigma^2(1+\exp\{4N/k\})}\right\rfloor$, we have

$$\begin{aligned} \log\frac{1}{\delta_{j^*}} &= \log\frac{(j^*)^2\pi^2}{6\delta} = \log\frac{1}{\delta} + \log\Big(\frac{\pi^2}{6}\Big) + 2\log(j^*) \\ &\le \log\frac{1}{\delta} + \log\frac{\pi^2}{6} + 2\log\Big(1 + k\log\frac{\sigma^2\big(1+\exp(4N/k)\big)}{\sigma^2+\|\theta\|^2/k}\Big) \\ &\le \log\frac{1}{\delta} + \log\frac{\pi^2}{6} + 2\log\Big(1 + k\log\big(1+\exp(4N/k)\big)\Big) \\ &\le \log\frac{1}{\delta} + \log\frac{\pi^2}{6} + 2\log\Big(1 + k\big(1+\frac{4N}{k}\big)\Big) \\ &\le \log\frac{1}{\delta} + \log\frac{\pi^2}{6} + \log(1+k+4N). \end{aligned}$$

Hence, the inequality

$$\begin{aligned}\mathcal{L}_{\mathcal{D}}\Big(\theta', \mathcal{N}(\theta, \sigma^2\mathbb{I}_k)\Big) &\leq \mathcal{L}_{\mathcal{S}}\Big(\theta', \mathcal{N}(\theta, \sigma^2\mathbb{I}_k)\Big) + \sqrt{\frac{\mathsf{KL}(Q\|P_{j^*}) + \log\frac{1}{\delta_{j^*}}}{2N}} \times L \\ &\leq \mathcal{L}_{\mathcal{S}}\Big(\theta', \mathcal{N}(\theta, \sigma^2\mathbb{I}_k)\Big) \\ &+ \frac{L}{2\sqrt{N}}\sqrt{1 + k\log\Big(1 + \frac{\|\theta\|^2}{k\sigma^2}\Big) + 2\log\frac{\pi^2}{6\delta} + 4\log(N+k)} \\ &\leq \mathcal{L}_{\mathcal{S}}\Big(\theta', \mathcal{N}(\theta, \sigma^2\mathbb{I}_k)\Big) \\ &+ \frac{L}{2\sqrt{N}}\sqrt{k\log\big(1 + \frac{\|\theta\|^2}{k\sigma^2}\big) + O(1) + 2\log\frac{1}{\delta} + 4\log(N+k)}.\end{aligned}$$

Since $\|\theta' - \theta\|^2$ is k chi-square distribution, for any positive t, we have

$$\mathbb{P}\big(\|\theta' - \theta\|^2 - k\sigma^2 \geq 2\sigma^2\sqrt{kt} + 2t\sigma^2)\big) \leq \exp(-t).$$

By choosing $t = \frac{1}{2}\log(N)$, with probability $1 - N^{-1/2}$, we have

$$\|\theta' - \theta\|^2 \leq \sigma^2\log(N) + k\sigma^2 + \sigma^2\sqrt{2k\log(N)} \leq k\sigma^2\Big(1 + \sqrt{\frac{\log(N)}{k}}\Big)^2.$$

By setting $\sigma = \rho \times \big(\sqrt{k} + \sqrt{\log(N)}\big)^{-1}$, we have $\|\theta' - \theta\|^2 \leq \rho^2$. Hence, we get

$$\begin{aligned}\mathcal{L}_{\mathcal{S}}\Big(\theta', \mathcal{N}(\theta, \sigma^2\mathbb{I}_k)\Big) &= \mathbb{E}_{\theta\sim\mathcal{N}(\theta,\sigma^2\mathbb{I}_k)}\mathbb{E}_{\mathcal{S}}\big[f_{\theta'}\big] = \int_{\|\theta'-\theta\|\leq\rho}\mathbb{E}_{\mathcal{S}}\big[f_{\theta'}\big]d\mathcal{N}(\theta, \sigma^2\mathbb{I}) \\ &+ \int_{\|\theta'-\theta\|>\rho}\mathbb{E}_{\mathcal{S}}\big[f_{\theta'}\big]d\mathcal{N}(\theta, \sigma^2\mathbb{I}) \\ &\leq \Big(1 - \frac{1}{\sqrt{N}}\Big)\max_{\|\theta'-\theta\|\leq\rho}\mathcal{L}_{\mathcal{S}}(\theta') + \frac{1}{\sqrt{N}}L \\ &\leq \max_{\|\theta'-\theta\|_2\leq\rho}\mathcal{L}_{\mathcal{S}}(\theta') + \frac{2L}{\sqrt{N}}.\end{aligned}$$

It follows that

$$\begin{aligned}\mathcal{L}_{\mathcal{D}}(\theta) &\leq \max_{\|\theta'-\theta\|\leq\rho}\mathcal{L}_{\mathcal{S}}(\theta') + \frac{4L}{\sqrt{N}}\Bigg[\sqrt{k\log\Big(1 + \frac{\|\theta\|^2}{\rho^2}\big(1 + \sqrt{\log(N)/k}\big)^2\Big)} \\ &+ 2\sqrt{\log\big(\frac{N+k}{\delta}\big) + O(1)}\Bigg] \\ &= \mathcal{L}_{\mathcal{D}}(\theta \mid \mathcal{S}) + \frac{4L}{\sqrt{N}}\Bigg[\sqrt{k\log\Big(1 + \frac{\|\theta\|^2}{\rho^2}\big(1 + \sqrt{\log(N)/k}\big)^2\Big)} \\ &+ 2\sqrt{\log\big(\frac{N+k}{\delta}\big) + O(1)}\Bigg].\end{aligned}$$

By choosing $\theta = \theta^*$, which is the solution on a random subset S^t and $\mathcal{S} = S^a$, which is another subset, $S^t, S^a \sim \mathcal{D}$, hence $N = N^a$, we reach the conclusion. □

Proof of Theorem 2

Proof. We have

$$\mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right) = \mathcal{L}_{B_t}\left(\theta_l\right) + \eta_1\|\nabla_\theta\mathcal{L}_{B^t}\left(\theta_l\right)\|_2^2 - \eta_2\nabla_\theta\mathcal{L}_{B^t}\left(\theta_l\right)\cdot\nabla_\theta\mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right).$$

This follows that

$$\begin{aligned}\nabla_\theta \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right) = {} & \nabla_\theta \mathcal{L}_{B_t}\left(\theta_l\right) + 2\eta_1 H_{B^t}\left(\theta_l\right) \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \\ & - \eta_2 \left[H_{B^t}\left(\theta_l\right) \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) + H_{B^a}\left(\tilde{\theta}_l^a\right) \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \right],\end{aligned}$$

where $H_{B^t}\left(\theta_l\right) = \nabla_\theta^2 \mathcal{L}_{B_t}\left(\theta_l\right)$ and $H_{B^a}\left(\tilde{\theta}_l^a\right) = \nabla_\theta^2 \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)$ are the Hessian matrices.

$$\begin{aligned}\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \cdot \nabla_\theta \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right) = {} & \nabla_\theta \mathcal{L}_{B_t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \\ & + 2\eta_1 \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^t}\left(\theta_l\right) \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \\ & - \eta_2 \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^t}\left(\theta_l\right) \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \\ & - \eta_2 \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^a}\left(\tilde{\theta}_l^a\right) \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right).\end{aligned}$$

We now choose $\eta_1 \leq \frac{\left|\nabla_\theta \mathcal{L}_{B_t}(\theta_l) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)\right|}{12\left|\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^t}(\theta_l) \nabla_\theta \mathcal{L}_{B^t}(\theta_l)\right|}$, we then have

$$\eta_1 \left| \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^t}\left(\theta_l\right) \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \right| \leq \frac{1}{12} \left| \nabla_\theta \mathcal{L}_{B_t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \right|.$$

This further implies

$$\eta_1 \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^t}\left(\theta_l\right) \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \geq -\frac{1}{12} \left| \nabla_\theta \mathcal{L}_{B_t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \right|.$$

Next we choose $\eta_2 \leq \min \left\{ \frac{\left|\nabla_\theta \mathcal{L}_{B_t}(\theta_l) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)\right|}{6\left|\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^t}(\theta_l) \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)\right|}, \frac{\left|\nabla_\theta \mathcal{L}_{B_t}(\theta_l) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)\right|}{6\left|\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^a}\left(\tilde{\theta}_l^a\right) \nabla_\theta \mathcal{L}_{B^t}(\theta_l)\right|} \right\}$, we then have

$$\eta_2 \left| \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^t}\left(\theta_l\right) \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \right| \leq \frac{\left| \nabla_\theta \mathcal{L}_{B_t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \right|}{6}.$$

$$-\eta_2 \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^t}\left(\theta_l\right) \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \geq -\frac{\left| \nabla_\theta \mathcal{L}_{B_t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \right|}{6}.$$

$$\eta_2 \left| \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^a}\left(\tilde{\theta}_l^a\right) \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \right| \leq \frac{\left| \nabla_\theta \mathcal{L}_{B_t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \right|}{6}.$$

$$-\eta_2 \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)^T H_{B^a}\left(\tilde{\theta}_l^a\right) \nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \geq -\frac{\left| \nabla_\theta \mathcal{L}_{B_t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \right|}{6}.$$

Finally, we yield

$$\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \cdot \nabla_\theta \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right) \geq \nabla_\theta \mathcal{L}_{B_t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) - \frac{1}{2} \left| \nabla_\theta \mathcal{L}_{B_t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right) \right|.$$

□

Proof of Theorem 3

Proof. We first denote $\hat{\theta}_l^t = \theta_l + \left(\rho_1 - \rho_2\right) \frac{\nabla \mathcal{L}_S(\theta_l)}{\|\nabla \mathcal{L}_S(\theta_l)\|}$. Using the β-smoothness, we have

$$\begin{aligned}\mathcal{L}_S\left(\theta_{l+1}\right) & \leq \mathcal{L}_S\left(\theta_l\right) + \nabla \mathcal{L}_S\left(\theta_l\right) \cdot \left(\theta_{l+1} - \theta_l\right) + \frac{\beta}{2}\|\theta_{l+1} - \theta_l\|^2 \\ & \leq \mathcal{L}_S\left(\theta_l\right) - \eta \nabla \mathcal{L}_S\left(\theta_l\right) \cdot \nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right) + \frac{\beta\eta^2}{2}\|\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)\|^2.\end{aligned}$$

Taking the expectation, we gain

$$
\begin{aligned}
\mathbb{E}\left[\mathcal{L}_S\left(\theta_{l+1}\right)\right] \leq & \mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\eta \mathbb{E}\left[\nabla \mathcal{L}_S\left(\theta_l\right) \cdot \nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)\right]+\frac{\beta \eta^2}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)\|^2\right] \\
\leq & \mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\eta \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|^2\right]-\eta \mathbb{E}\left[\nabla \mathcal{L}_S\left(\theta_l\right) \cdot\left[\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)-\nabla \mathcal{L}_S\left(\theta_l\right)\right]\right] \\
& +\frac{\beta \eta^2}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)\|^2 \mid\right] \\
= & \mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\eta \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|^2\right]-\eta \mathbb{E}\left[\nabla \mathcal{L}_S\left(\theta_l\right) \cdot\left[\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)-\nabla \mathcal{L}_{B^t}\left(\hat{\theta}_l^t\right)\right]\right] \\
& +\frac{\beta \eta^2}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)\|^2 \mid\right]-\eta \mathbb{E}\left[\nabla \mathcal{L}_S\left(\theta_l\right) \cdot\left[\nabla \mathcal{L}_S\left(\hat{\theta}_l^t\right)-\nabla \mathcal{L}_S\left(\theta_l\right)\right]\right] \\
\leq & \mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\eta \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|^2\right] \\
& +\frac{\eta}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|^2\right]+\frac{\eta}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)-\nabla \mathcal{L}_{B^t}\left(\hat{\theta}_l^t\right)\|^2\right] \\
& -\eta \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\| \frac{\hat{\theta}_l^t-\theta_l}{\rho_1-\rho_2} \cdot\left[\nabla \mathcal{L}_S\left(\hat{\theta}_l^t\right)-\nabla \mathcal{L}_S\left(\theta_l\right)\right]\right]+\frac{\beta \eta^2}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)\|^2\right] \\
\leq & \mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\frac{\eta}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|^2\right]+\frac{\beta^2 \eta}{2} \mathbb{E}\left[\|\tilde{\theta}_l^t-\hat{\theta}_l^t\|^2\right]+\frac{\beta \eta^2}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)\|^2\right] \\
& +\beta \eta \mathbb{E}\left[\frac{\|\nabla \mathcal{L}_S\left(\theta_l\right)\|}{\rho_1-\rho_2}\|\hat{\theta}_l^t-\theta_l\|^2\right] \\
\leq & \mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\frac{\eta}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|^2\right] \\
& +\frac{\beta^2 \eta}{2} \mathbb{E}\left[\|\rho_1 \frac{\nabla \mathcal{L}_{B^t}\left(\theta_l\right)}{\|\nabla \mathcal{L}_{B^t}\left(\theta_l\right)\|}-\rho_2 \frac{\nabla \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)}{\|\nabla \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)\|}-\left(\rho_1-\rho_2\right) \frac{\nabla \mathcal{L}_S\left(\theta_l\right)}{\|\nabla \mathcal{L}_S\left(\theta_l\right)\|}\|^2\right] \\
& +\frac{\beta \eta^2}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)\|^2\right]+\beta \eta \mathbb{E}\left[\frac{\|\nabla \mathcal{L}_S\left(\theta_l\right)\|}{\rho_1-\rho_2}\|\hat{\theta}_l^t-\theta_l\|^2\right] \\
\leq & \mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\frac{\eta}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|^2\right]+3 \beta^2 \eta\left(\rho_1^2+\rho_2^2-\rho_1 \rho_2\right) \\
& +\left(\rho_1-\rho_2\right) \beta \eta \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|\right]+\frac{\beta \eta^2}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)\|^2\right] \\
\leq & \mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\frac{\eta}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|^2\right]+3 \beta^2 \eta\left(\rho_1^2+\rho_2^2-\rho_1 \rho_2\right) \\
& +\left(\rho_1-\rho_2\right) \beta \eta \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|\right] \\
& +\beta \eta^2 \mathbb{E}\left[\|\nabla \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)-\nabla \mathcal{L}_S\left(\tilde{\theta}_l^t\right)\|^2\right]+\beta \eta^2 \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\tilde{\theta}_l^t\right)\|^2\right] \\
\leq & \mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\frac{\eta}{2} \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|^2\right]+3 \beta^2 \eta\left(\rho_1^2+\rho_2^2-\rho_1 \rho_2\right) \\
& +\left(\rho_1-\rho_2\right) \beta \eta \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|\right]+\beta \eta^2\left(\sigma^2+G^2\right) \\
& \leq \mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\frac{\eta}{2}\left(\mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|\right]-\Delta \rho \beta\right)^2+\frac{1}{2} \eta \Delta \rho^2 \beta^2 \\
& +3 \beta^2 \eta\left(\rho_1^2+\rho_2^2-\rho_1 \rho_2\right)+\beta \eta^2\left(\sigma^2+G^2\right)
\end{aligned}
$$

Rearrange the terms, we obtain

$$
\begin{aligned}
\left(\mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|\right]-\Delta \rho \beta\right)^2 \leq & \frac{2}{\eta}\left(\mathbb{E}\left[\mathcal{L}_S\left(\theta_l\right)\right]-\mathbb{E}\left[\mathcal{L}_S\left(\theta_{l+1}\right)\right]\right) \\
& +\frac{1}{2} \eta \Delta \rho^2 \beta^2+3 \beta^2 \eta\left(\rho_1^2+\rho_2^2-\rho_1 \rho_2\right)+\beta \eta^2\left(\sigma^2+G^2\right)
\end{aligned}
$$

Take sum l from 1 to T, we reach

$$\begin{aligned}\frac{1}{T}\sum_{l=1}^{T}\left(\mathbb{E}\left[\|\nabla\mathcal{L}_S\left(\theta_l\right)\|\right]-\Delta\rho\beta\right)^2 \leq & \frac{2}{\eta T}\left(\mathbb{E}\left[\mathcal{L}_S\left(\theta_0\right)\right]-\mathbb{E}\left[\mathcal{L}_S\left(\theta_{T+1}\right)\right]\right)+\Delta\rho^2\beta^2 \\ & +6\beta^2\left(\rho_1^2+\rho_2^2-\rho_1\rho_2\right)+2\beta\eta\left(\sigma^2+G^2\right) \\ \leq & \frac{2}{\eta T}\left(\mathbb{E}\left[\mathcal{L}_S\left(\theta_0\right)\right]-L^*\right)+\Delta\rho^2\beta^2+6\beta^2\left(7\rho_1^2+7\rho_2^2-8\rho_1\rho_2\right) \\ & +2\beta\eta\left(\sigma^2+G^2\right) \\ \leq & \frac{2\Delta}{\eta T}+6\beta^2\left(7\rho_1^2+7\rho_2^2-8\rho_1\rho_2\right)+2\beta\eta\left(\sigma^2+G^2\right)\end{aligned}$$

Substitute $\eta=\frac{\sqrt{\Delta}}{\sqrt{\beta T(\sigma^2+G^2)}}$, we arrive at

$$\frac{1}{T}\sum_{l=1}^{T}\left(\mathbb{E}\left[\|\nabla\mathcal{L}_S\left(\theta_l\right)\|\right]-\Delta\rho\beta\right)^2 \leq \frac{4\sqrt{\beta\Delta\left(\sigma^2+G^2\right)}}{\sqrt{T}}+6\beta^2\left(7\rho_1^2+7\rho_2^2-8\rho_1\rho_2\right)$$

□

Proof of Theorem 3.1

Proof. We have

$$\begin{aligned}\min_{l=0,\ldots,T}\left|\mathbb{E}\left[\|\nabla\mathcal{L}_S\left(\theta_l\right)\|\right]-\Delta\rho\beta\right| \leq & \frac{1}{T}\sum_{l=1}^{T}\left|\mathbb{E}\left[\|\nabla\mathcal{L}_S\left(\theta_l\right)\|\right]-\Delta\rho\beta\right| \\ \leq & \sqrt{\frac{1}{T}\sum_{l=1}^{T}\left(\mathbb{E}\left[\|\nabla\mathcal{L}_S\left(\theta_l\right)\|\right]-\Delta\rho\beta\right)^2} \\ \leq & \frac{2\left[\beta\Delta\left(\sigma^2+G^2\right)\right]^{1/4}}{T^{1/4}}+\beta\sqrt{6\left(7\rho_1^2+7\rho_2^2-8\rho_1\rho_2\right)}\end{aligned}$$

This implies that

$$\begin{aligned}\min_{l=0,\ldots,T}\mathbb{E}\left[\|\nabla\mathcal{L}_S\left(\theta_l\right)\|\right] \leq & \frac{2\left[\beta\Delta\left(\sigma^2+G^2\right)\right]^{1/4}}{T^{1/4}}+\beta\sqrt{6\left(7\rho_1^2+7\rho_2^2-8\rho_1\rho_2\right)} \\ & +\Delta\rho\beta.\end{aligned}$$

□

A.2 Convergence analysis

To do convergence analysis for Align-SAM, we make the following assumptions (Si & Yun, 2023):

A1 (G-Lipchitz). The loss function $\mathcal{L}_S$ is G-Lipchitz, i.e., $|\mathcal{L}_S(\theta)-\mathcal{L}_S(\theta')|\leq G\|\theta-\theta'\|$.

A2 (β-smoothness). The loss function $\mathcal{L}_S$ is β-smooth, if $\|\nabla\mathcal{L}_S\left(\theta\right)-\nabla\mathcal{L}_S\left(\theta'\right)\|\leq\beta\|\theta-\theta'\|$ for all θ,θ'.

A3 (Bounded variance). For any batch $B\sim S$, $\mathbb{E}_B\left[\|\mathcal{L}_B\left(\theta\right)-\mathcal{L}_S\left(\theta\right)\|^2\right]\leq\sigma^2$ for all θ.

Theorem 3. *Assume that the loss function $\mathcal{L}_S$ satisfies the assumptions A1,A2, and A3, and $L^*=inf_\theta\mathcal{L}_S(\theta)>-\infty$. Under Align-SAM, starting from θ_0 and the learning rate $\eta=\frac{\sqrt{\Delta}}{\sqrt{\beta T(\sigma^2+G^2)}}$, we have*

$$\frac{1}{T}\sum_{l=1}^{T}\left(\mathbb{E}\left[\|\nabla\mathcal{L}_S\left(\theta_l\right)\|\right]-\Delta\rho\beta\right)^2 \leq \frac{4\sqrt{\beta\Delta\left(\sigma^2+G^2\right)}}{\sqrt{T}}+6\beta^2\left(7\rho_1^2+7\rho_2^2-8\rho_1\rho_2\right),$$

where $\Delta\rho=\rho_1-\rho_2$ and $\Delta=\mathcal{L}_S(\theta_0)-L^*$.

Table 6: Domain Generalization setting. All models are trained on ImageNet-1k and then evaluated on ImageNet-1k (clean validation set), Imagenet-R, and Imagenet-C datasets.

Method	Top-1 Acc			Top-5 Acc		
	Imagenet	ImageNet-R	ImageNet-C	Imagenet	ImageNet-R	ImageNet-C
ResNet18 - Transfer Learning						
SAM	70.52	34.18	48.27	89.60	52.82	72.17
Align-SAM	**70.88**	**34.38**	**48.69**	**89.94**	**53.23**	**72.61**
ResNet18 - From Scratch						
SAM	62.46	25.86	32.96	73.15	43.09	55.72
Align-SAM	**63.64**	**26.20**	**34.06**	**73.45**	**43.99**	**57.42**

Corollary 3.1. *Under the assumptions as in Theorem 3, we have*

$$\min_{l=0,\ldots,T} \mathbb{E}\left[\|\nabla \mathcal{L}_S\left(\theta_l\right)\|\right] \leq \frac{2\left[\beta\Delta\left(\sigma^2+G^2\right)\right]^{1/4}}{T^{1/4}} + \beta\left(\sqrt{6\left(7\rho_1^2+7\rho_2^2-8\rho_1\rho_2\right)}+\Delta\rho\right).$$

It is well known that the normalized (practical) version of SAM *does not converge* to the minimizer of the training loss, as rigorously demonstrated in Si & Yun (2023) (Theorem 4.6), one of the most comprehensive analyses of SAM's convergence behavior. Our proposed approach shares the *same convergence rate* as standard SAM, as established in Si & Yun (2023) (Theorem 4.6).

A.3 Additional Experiments

Experiments of Domain Generalization To strengthen the claims about Align-SAM, we evaluated Align-SAM's robustness under domain shifts by training on ImageNet-1K and testing on ImageNet-1K (clean test set), ImageNet-R (artistic renditions), and ImageNet-C (with various corruptions). These shifts demonstrate Align-SAM's robustness, as it consistently outperforms SAM across all setups. The results are shown in Table 6.

Experiments of meta-learning setting The concept of Align-SAM is inspired by the agnostic approach in the MAML setting, where the meta-model is optimized on the meta-training set but aims to minimize loss on the validation set, assuming both the training and validation sets are from the same data distribution. Different from this original idea, Align-SAM uses the gradient from another auxiliary set as an indicator to close the generalization gap between the training and testing sets. Despite this difference, both approaches share the same underlying objective, making it reasonable to expect that applying Align-SAM in the MAML setting will result in improved generalization performance.

Table 7: Meta-learning results on Mini-Imagenet dataset. All baseline results are taken from Abbas et al. (2022)

Method	Accuracy	
	5 ways 1 shot	5 ways 5 shots
MAML	47.13	62.20
SHARP-MAML$_{low}$	49.72	63.18
Align-SAM$_{low}$	**50.08**	**64.29**

We compare our approach with standard MAML and Sharp-MAML (Abbas et al., 2022), which also address the loss landscape flatness in bilevel models. MAML is typically framed as a bilevel optimization problem, consisting of a meta-update step to learn a shared model initialization and a fine-tuning step to adapt task-specific models. Sharp-MAML analyzes the geometry of MAML's loss landscape and introduces the use of SAM to avoid sharp local minima in MAML loss functions. Sharp-MAML proposes three variants: Sharp-MAML$_{low}$ (applying SAM only to the fine-tuning step), Sharp-MAML$_{up}$ (applying SAM only to the meta-update step), and Sharp-MAML$_{both}$ (applying SAM to both steps). Since our Align-SAM shares the same objective as SAM to improve model

Table 8: Meta-learning results on Omniglot dataset. All baseline results are taken from Abbas et al. (2022)

Method	Accuracy	
	20 ways 1 shot	20 ways 5 shots
MAML	91.77	96.16
SHARP-MAML$_{low}$	**92.89**	96.59
Align-SAM$_{low}$	92.66	**97.28**

generalization it can replace SAM optimization in both the meta-update and fine-tuning steps of the MAML model.

The experiments follow the setup from Abbas et al. (2022), specifically using the Sharp-MAML$_{low}$ variation, which focuses on minimizing the sharpness of meta-models fine-tuned on the meta-training set. For Align-SAM, we set $\lambda = 2$ and follow the setup exactly for ρ, inner gradient steps, and step size in Abbas et al. (2022) for both Sharp-MAML$_{low}$ and Align-SAM$_{low}$. The results are reported in Tables 7 and 8. Our method outperforms most baselines with significant improvements, demonstrating the effectiveness of Align-SAM and its flexibility across various settings.

Cosine similarity of gradients In Theorem 2, we prove that minimizing the loss function $\mathcal{L}_{B^t}$ could encourage two gradients $\nabla_\theta \mathcal{L}_{B^t}\left(\tilde{\theta}_l^t\right)$ and $\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)$ to be more congruent since our update aims to maximize its lower bound, which is $\nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)$. In this section, we measure the cosine similarity between two gradients $\nabla_\theta \mathcal{L}_{B^t}\left(\theta_l\right)$ and $\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)$ before (denoted as $cosine_b$ in Figure 1a) and after (denoted as $cosine_a$in Figure 1b) updating the model and measure the change of these two score (denoted as $change$). Mathematically, we define $cosine_b = \frac{\nabla_\theta \mathcal{L}_{B^t}(\theta_l) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)}{\|\nabla_\theta \mathcal{L}_{B^t}(\theta_l)\|_2 \|\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_l^a\right)\|_2}$, $cosine_a = \frac{\nabla_\theta \mathcal{L}_{B^t}(\theta_{l+1}) \cdot \nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_{l+1}^a\right)}{\|\nabla_\theta \mathcal{L}_{B^t}(\theta_{l+1})\|_2 \|\nabla_\theta \mathcal{L}_{B^a}\left(\tilde{\theta}_{l+1}^a\right)\|_2}$, and $change = \frac{cosine_a - cosine_b}{cosine_a}$.

As shown in Figure 1c, both SAM and Align-SAM improve the similarity after updating the model, this improvement also increases across training epochs. However, the similarity score of our Align-SAM is always higher than SAM across the training process, both before and after updating the model. It is evident that our Align-SAM encourages gradients of the training subset and the auxiliary subset to be more similar during the training process.

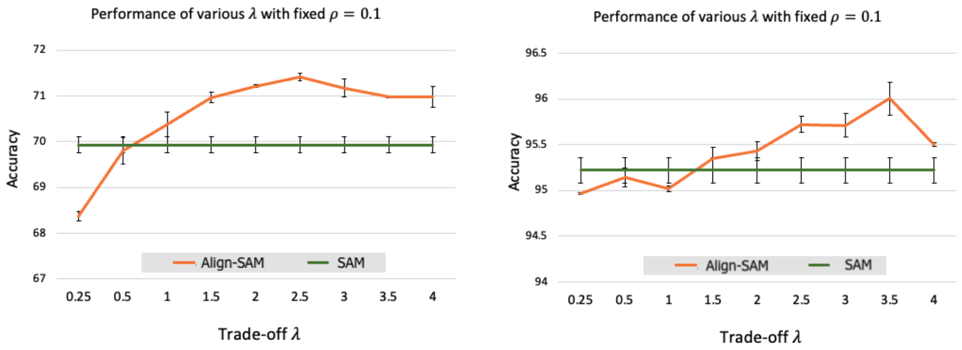
Batch size $|B^a|$ and complexity Our method is to use a gradient on the auxiliary subset as a helper indicator to lead the model to wider local minima while maintaining low loss on a random subset from the same distribution, and the model should be updated mainly using training samples. Increasing the batch size $|B^a|$ could potentially increase performance and training time. In Table 9, we present the results of Align-SAM with various sizes $|B^a|$ of CIFAR-100 with Resnet32 while maintaining a fixed training batch size $|B^t| = 512$. We consider performance and training complexity to be the trade-off of Align-SAM and find that setting $|B^t| = 4|B^a|$ works well for all experiments.

Additionally, we examine different numbers of epochs with SAM, Align-SAM, and LookBehind-SAM Mordido et al. (2024). As presented in Table 9, Align-SAM outperforms both SAM and Lookbehind-SAM under the same training time.

Sensitivity of trade-off coefficient λ Throughout this paper, we used a consistent setting of $\lambda = 2$, which is the trade-off coefficient for combining gradients from B^t and B^a. While this hyperparameter could be optimized for each experiment individually, we find that this configuration delivers good performance across most experiments. By setting $\lambda > 1$, we ensure that the perturbed model prioritizes maximizing the loss on the training mini-batch B^t rather than minimizing it on the auxiliary mini-batch B^a. This approach encourages the model to focus primarily on minimizing sharpness during the actual update step in Formula 6.

Table 9: Experiments on different sizes of auxiliary mini-batch with a fixed size of training mini-batch is 512 samples

Method	Epoch	Auxiliary batch-size	Accuracy	Training time	
				per epoch	Total
SAM	200	-	70.31 ± 0.233	11s	36.6m
	220		70.45 ± 0.303	11s	40.3m
	250		70.06 ± 0.078	11s	45.8m
	270		71.15 ± 0.293	11s	49.5m
Align-SAM	200	16	70.58 ± 0.219	11s	37.5m
		32	71.07 ± 0.172	12s	40.0m
		64	70.67 ± 0.049	13s	43.3m
		128	71.21 ± 0.056	14s	46.6m
		256	71.04 ± 0.207	15s	50.0m
SAM	220	-	70.45 ± 0.303	11s	40.3m
Align-SAM	170	128	70.83 ± 0.209	14s	39.6m



(a) Experiments with CIFAR-100 on Resnet32 (b) Experiments with Flower102 on EfficientNet-B2

Figure 2: Experiments of various trade-off λ with fixed perturbation radius ρ

To verify the impact of this hyperparameter on model performance, we conduct experiments with varying values of trade-off λ and present the results in Figure 2. Notably, the configuration where $\lambda \geq 1$ consistently yields higher accuracy compared to the setting where $\lambda < 1$. When decreasing λ, the model places more emphasis on minimizing the auxiliary loss, rather than sharpness on the training set during the actual update step in Formula 6, ultimately reducing performance.

Analysis of loss landscape and eigenvalues of the Hessian matrix We demonstrate the effectiveness of Align-SAM in guiding models toward flatter regions of the loss landscape, as compared to both SAM and SGD, in Figures 3 and 4. The loss landscapes are visualized with the same setting, the blue areas represent lower loss values, while the red areas indicate higher loss values. Although SAM is shown to lead the model to a flatter region than SGD, Align-SAM achieves an even smoother and significantly flatter loss landscape, especially in experiments with EfficientNet-B2 in Figure 3.

To further validate that Align-SAM successfully locates minima with low curvature, we compute the Hessian of the loss landscape and report the five largest eigenvalues, sorted from λ_1 to λ_5, in Table 10. These eigenvalues provide insight into the curvature of the model at the optimized parameters. Larger eigenvalues indicate steeper curvature, meaning the model is more sensitive to small changes in its parameters. Conversely, smaller eigenvalues suggest flatter minima, which are typically associated with improved robustness, better generalization, and reduced sensitivity to overfitting. Negative eigenvalues indicate non-convex curvature in certain directions.

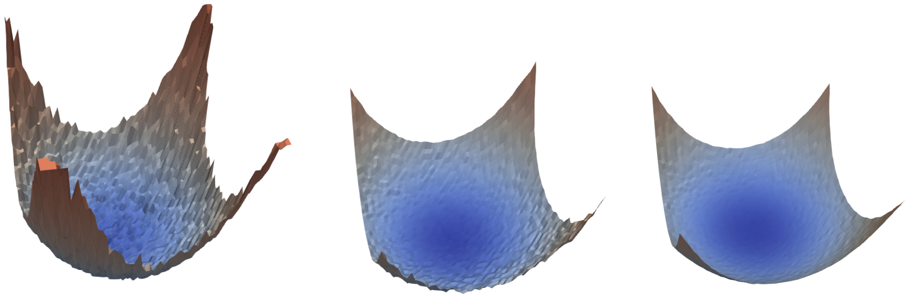


Figure 3: Loss landscape of **EffecientNet-B2** trained on Flower102 dataset with **(left)** SGD, **(middle)** SAM, and **(right)** Align-SAM.

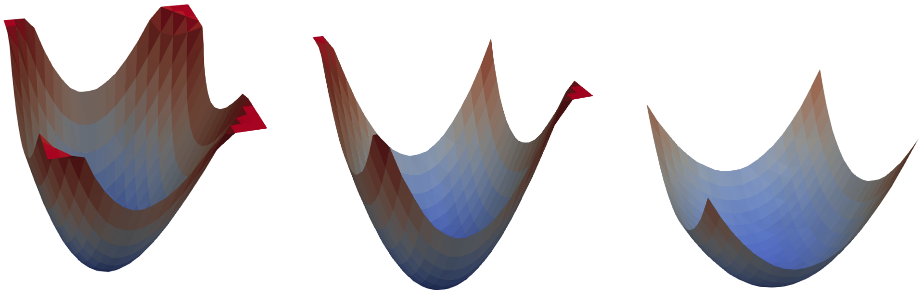


Figure 4: Loss landscape of **ResNet32** trained **(left)** SGD, **(middle)** SAM, and **(right)** Align-SAM on Cifar100 dataset.

Methods	Ratio of top-5 largest eigenvalues of Hessian matrix				
	λ_1	$\frac{\lambda_1}{\lambda_2}$	$\frac{\lambda_1}{\lambda_3}$	$\frac{\lambda_1}{\lambda_4}$	$\frac{\lambda_1}{\lambda_5}$
EfficientNet-B2 on Flower102					
SGD	2.05×10^5	4.55	7.88	-4.36	-4.18
SAM	1.61×10^3	1.27	1.31	1.54	-1.65
Align-SAM	0.61×10^3	1.48	1.64	1.90	1.96
Resnet32 on Cifar100					
SGD	3.07×10^5	1.27	1.46	1.87	2.13
SAM	1.42×10^5	1.47	1.65	1.79	1.89
Align-SAM	1.11×10^5	1.32	1.68	1.85	2.09

Table 10: Eigenvalues of Hessian matrix

As shown in Table 10, Align-SAM consistently achieves positive and lower eigenvalues compared to the baseline methods, suggesting that it effectively leads the model toward flatter regions of the loss landscape. These results further support the efficacy of Align-SAM in optimizing for smoother and more stable solutions across a variety of architectures and tasks.

A.4 Ethical Statement and The Use of Large Language Models

We used a large language model (ChatGPT) to help with editing this paper. It was only used for simple tasks such as fixing typos, rephrasing sentences for clarity, and improving word choice. All ideas, experiments, and analyses were done by the authors, and the use of LLMs does not affect the reproducibility of our work.