RSAM: LEARNING ON MANIFOLDS WITH RIEMANNIAN SHARPNESS-AWARE MINIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Nowadays, understanding the geometry of the loss landscape shows promise in enhancing a model's generalization. In our work, we draw upon prior research that applies geometric principles to optimization and present a novel approach to improve robustness and generalization ability for constrained optimization problems. This paper aims to generalize the Sharpness-Aware Minimization (SAM) optimizer to Riemannian manifolds. In doing so, we extend the concept of sharpness and introduce a novel notion of sharpness on manifolds. To support this notion of sharpness, we present a theoretical analysis characterizing generalization capabilities with respect to manifold sharpness, which demonstrates a tighter bound on the generalization gap, a result not known before. Motivated by this analysis, we introduce our algorithm, Riemannian Sharpness-Aware Minimization (RSAM). To demonstrate RSAM's ability to enhance model generalization, we evaluate and contrast our algorithm on a broad set of problems, such as image classification and contrastive learning across different datasets, including CIFAR100, CIFAR10, and FGVCAircraft.

1 INTRODUCTION

One of the challenges in deep learning is the overfitting issue, which is attributed to high-dimensional and non-convex loss functions, which leads to complicated loss landscapes with multiple local minima. Hence, it is crucial to understand the geometry of the loss landscape to train robust models and improve generalization ability. Regarding this issue, flat minimizes, which seek for regions with low sharpness, have been known to be among the most effective approaches for robustness (Keskar et al., 2016; Kaddour et al., 2022b; Li et al., 2022). Indeed, Sharpness-Aware Minimization (SAM), as introduced by Foret et al. (2021b), stands out as a notable method by simultaneously minimizing the loss function and the worst-case loss within a neighborhood of the current model. Nowadays, SAM has proven to be versatile across a diverse array of tasks such as meta-learning (Abbas et al., 2022), federated learning (Qu et al., 2022), vision models (Chen et al., 2021), or language models (Bahri et al., 2022).

Another challenge within deep learning is that to preserve the model's robustness, it is often desirable to impose strict constraints on the parameters. Such constraints include the SPD constraints (Gao et al., 2020), orthogonality, and full rank (Xie et al., 2017; Roy et al., 2019; Wang et al., 2020). In such cases, the models are known to reside on some Riemannian manifolds, such as Grassmann manifolds, SPD manifolds, or Stiefel manifolds. Given the importance of understanding the geometry of the parameter space, especially when it is a differential manifold, various optimization techniques have been developed to learn on Riemannian manifolds (Bonnabel, 2013; Luenberger, 1972; Kasai et al., 2019; Sato et al., 2019; Zhang et al., 2017). Indeed, prior studies have demonstrated that taking into account this intrinsic geometry structure will remarkably improve the model's generalization ability (Roy et al., 2019; Absil et al., 2008a).

Understanding these two challenges in enhancing the model's robustness and generalization ability, we seek to generalize the SAM optimizer to Riemannian manifolds. In particular, we introduce a novel notion of sharpness on manifolds to study the intrinsic geometry of the parameter space and the loss landscape. This notion is backed by a comprehensive theoretical analysis that formulates generalization capacity in terms of neighborhood-wise training loss on the manifolds Indeed, our theorem establishes a tighter upper bound of $\mathcal{O}(\sqrt{d})$ compared to the existing bounds such as

$\mathcal{O}(\sqrt{k})$ from Foret et al. (2021b), in which d is the dimensionality of the manifold, embedded in a higher dimensional Euclidean space with $k \gg d$ dimensions. Motivated by this theoretical analysis, we propose a Riemannian optimization technique called Riemannian Sharpness-Aware Minimization (RSAM). We show via an empirical study that RSAM improves the model's generalization ability across a range of different tasks such as supervised learning, self-supervised learning, and a diverse array of computer vision datasets (CIFAR100, CIFAR10, FGVCAircraft), as well as different models (ResNet34, ResNet50). Indeed, RSAM makes a notable improvement upon SAM and SupCon. We will also show in our ablation studies the ability of RSAM to seek flat regions on the loss landscape with the aid of Riemannian geometry. In short, our contributions are as follows:

- We introduce a novel notion of Sharpness on Riemannian manifolds, backed by a theoretical analysis establishing a tighter upper bound than the existing bounds.
- Motivated by the theory, we introduce RSAM and empirically study its efficacy across various settings. Our experiments show that RSAM outperforms current methods by notable margins.

2 Related Works

2.1 Sharpness Aware Minimization

The Sharpness-Aware Minimization (SAM) technique, introduced by Foret et al. (2021a), has gained prominence due to its effectiveness and scalability compared to previous methods. SAM's versatility is evident across various tasks and domains, making it a powerful optimization approach. SAM has found applications in diverse areas such as meta-learning bi-level optimization (Abbas et al., 2022), federated learning (Qu et al., 2022), vision models (Chen et al., 2021), language models (Bahri et al., 2022), domain generalization (Cha et al., 2021), and multi-task learning (Phan et al., 2022).

Recent research has further enhanced SAM's capabilities by exploring its underlying geometry (Kwon et al., 2021; Kim et al., 2022), minimizing surrogate gaps (Zhuang et al., 2022), and speeding up training time (Du et al., 2022; Liu et al., 2022). Additionally, Kaddour et al. (2022a) empirically studied SAM's sharpness compared to SWA (Izmailov et al., 2018). In contrast, BSAM (Möllenhoff & Khan, 2023) demonstrated that SAM is an optimal Bayesian relaxation of standard Bayesian inference with a normal posterior. Moreover, Nguyen et al. (2023b) developed the sharpness concept for Bayesian Neural Networks. Finally, Nguyen et al. (2023a) generalized SAM by leveraging optimal transport based distributional robustness with sharpness-aware minimization.

2.2 Learning on Manifold

Within the literature of machine learning, it is often desirable to impose constraints on a model's parameters of a model, such as orthogonality or full rank. In such cases, the search space is no longer an Euclidean space but a manifold. Studies in Riemannian geometry have indicated that considering the intrinsic geometry of the parameter space during training can yield better performance (Roy et al., 2019; Absil et al., 2008a). Indeed, the awareness of the intrinsic geometry can increase the likelihood of discovering desirable parameters and optimize training time by confining the search to a significantly lower-dimensional space.

As such, various classes of manifolds have been introduced across a spectrum of applications. For example, in the domain of metric learning, Roy et al. (2019) incorporated Stiefel manifolds to ensure that the learned parameters maintain orthogonality constraints. In the context of Gaussian mixture models, Gao et al. (2020) proposed a strategy involving learning on SPD manifolds to enforce SPD constraints. Furthermore, Grassmann manifolds have found applications in diverse domains, encompassing recommender systems (Dai et al., 2012; Boumal & Absil, 2015) or modeling affine subspaces within document-specific language models (Hall & Hofmann, 2000).

Various procedures to learn on Riemannian manifolds were proposed. One notable approach to enforce adherence to these manifold structures is the Riemannian gradient descent (RGD) algorithm (Luenberger, 1972). However, it is worth noting that RGD has computational limitations since it requires taking the whole dataset for each iteration. To address this issue, Bonnabel (2013) introduced the Riemannian stochastic gradient descent (RSGD) method, which reduces computational overhead and thus gains widespread adoption and is used on various manifolds such as SPD manifolds.

3 RSAM: RIEMANNIAN SHARPNESS-AWARE MINIMIZATION

3.1 FORMULATIONS AND NOTATIONS

This section presents the problem formulations and the notions used in our theory development. We consider a model $f_{\boldsymbol{\theta}} : X \to Y$ parameterized by parameter $\boldsymbol{\theta}$, where X is the data space and Y is the label space. Let $\mathcal{D}$ be the data/label distribution that generates data/label pair $(\mathbf{x}, \mathbf{y}) \sim \mathcal{D}$. Based on $\mathcal{D}$, we sample a specific training set $\mathcal{S} = \{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \cdots, (\mathbf{x}_n, \mathbf{y}_n)\}$, where $(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_n, \mathbf{y}_n) \overset{\text{iid}}{\sim} \mathcal{D}$.

Given $(x, y) \sim \mathcal{D}$, we use the per-sample loss function $\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), \mathbf{y})$ to quantify the loss suffered by the model $f_{\boldsymbol{\theta}}$ when predicting $(\mathbf{x}, \mathbf{y})$. At such, the empirical loss on the training set $\mathcal{S}$ is $\mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} \ell(f_{\boldsymbol{\theta}}(\mathbf{x}_i), \mathbf{y}_i)$, while the general loss on the data/label distribution $\mathcal{D}$ is $\mathcal{L}_{\mathcal{D}}(\boldsymbol{\theta}) = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\big[\ell(f_{\boldsymbol{\theta}}(\mathbf{x}), \mathbf{y})\big]$.

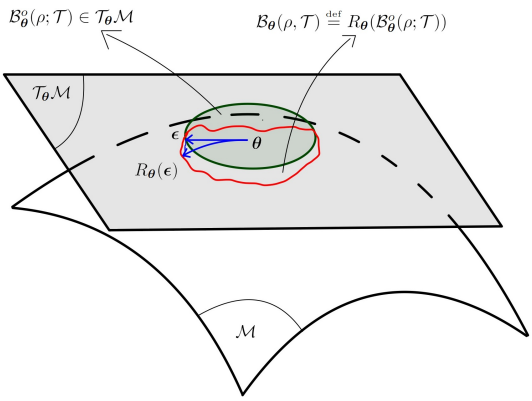


Figure 1: Neighborhoods on manifolds. $\mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{T})$ (green) is an ρ-ball on the tangent space $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$. The neighborhood $\mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M})$ (red) of $\boldsymbol{\theta}$ on $\mathcal{M}$ is the retraction image of $\mathcal{B}^o_{\boldsymbol{\theta}}(\rho;\mathcal{T})$.

In this work, we assume that some constraints are imposed on the models (e.g., orthogonality, full rank, or SPD constraints), making the model parameters $\boldsymbol{\theta}$ lying in a *low-dimensional manifold* $\mathcal{M} \subset \mathbb{R}^k$ embedded in the *ambient vector space* $\mathbb{R}^k$. In other words, we assume that $\boldsymbol{\theta} \in \mathcal{M}$, in which $\mathcal{M}$ has dimensionality $d \ll k$. Bearing this in mind, we further develop Riemannian Sharpness-aware Minimization in the next section. For our theory development, we also introduce a few additional relevant notions. Indeed, for a point $\boldsymbol{\theta} \in \mathcal{M}$, we denote $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$ as the tangent space of $\mathcal{M}$. By convention, the tangent space $\mathcal{T}_{\theta}\mathcal{M}$ uses the coordinate system with the current $\boldsymbol{\theta}$ as an origin hence $\boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$ specifies the offset from $\boldsymbol{\theta}$, thus further representing the vector $\boldsymbol{\theta} + \boldsymbol{\epsilon}$ on the ambient vector space $\mathbb{R}^k$. On the tangent space, we need to define the neighborhood (e.g., the ρ-ball) around $\boldsymbol{\theta}$ for presenting the concept of sharpness over the tangent space and the manifold. To serve our theory development, we define two equivalent ρ-balls w.r.t. the tangent and ambient space coordinate systems: $\mathcal{B}^o_{\boldsymbol{\theta}}(\rho;\mathcal{T}) = \{\boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M} : \|\boldsymbol{\epsilon}\|_2 \leq \rho\}$ (i.e., the offset ball) and $\mathcal{B}^a_{\boldsymbol{\theta}}(\rho;\mathcal{T}) = \boldsymbol{\theta} + \mathcal{B}^o_{\boldsymbol{\theta}}(\rho;\mathcal{T}) = \{\boldsymbol{\theta} + \boldsymbol{\epsilon} : \boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M} \text{ and } \|\boldsymbol{\epsilon}\|_2 \leq \rho\}$ (i.e., the absolute ball). Furthermore, we define the neighborhood of $\boldsymbol{\theta}$ on the manifold $\mathcal{M}$ as the retraction map of the one on the tangent space: $\mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M}) = R_{\boldsymbol{\theta}}(\mathcal{B}^o_{\boldsymbol{\theta}}(\rho;\mathcal{T}))$, where $R_{\boldsymbol{\theta}}$ specifies the retraction operation at $\boldsymbol{\theta}$. For clarity, we refer to Figure 1 for an illustration of the neighborhood on Riemannian manifolds.

3.2 OUR THEORY DEVELOPMENT

In what follows, we present our theory development for RSAM. We consider the minimization problem in which the parameter space is an embedded submanifold $\mathcal{M} \in \mathbb{R}^k$:

$$\min_{\boldsymbol{\theta}\in\mathcal{M}} \mathcal{L}_{\mathcal{D}}(\boldsymbol{\theta})$$

Here, we can think of the condition $\boldsymbol{\theta} \in \mathcal{M}$ as a constraint to the optimization problem, such as orthogonality. There are two significant challenges regarding this constrained optimization problem. The first challenge is that $\mathcal{D}$ exists but is unknown and can only be realized through the training set $\mathcal{S}$. The second challenge, which is our main concern, is that $\boldsymbol{\theta}$ must remain within a manifold $\mathcal{M}$, i.e., satisfy the constraints. To tackle the second challenge, we present our main theorem about generalization ability on manifolds whose proof can be found in Appendix A.1.3.

Theorem 1. *For any small $\rho > 0$ and $\delta \in [0; 1]$, with a high probability $1 - \delta$ over training set $\mathcal{S}$ generated from a distribution $\mathcal{D}$, we have the following inequalities on the tangent space $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$ and the manifold $\mathcal{M}$:*

i) The upper-bound on the tangent space $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$:

$$\mathcal{L}_{\mathcal{D}}(\boldsymbol{\theta}) \leq \max_{\boldsymbol{\theta}' \in \mathcal{B}^a_{\boldsymbol{\theta}}(\rho;\mathcal{T})} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}') + \mathcal{O}\left(\sqrt{\frac{\frac{d}{\rho^2} + d + \log\frac{n}{\delta}}{n-1}}\right) \tag{1}$$

ii) The upper-bound on the manifold $\mathcal{M}$:

$$\mathcal{L}_{\mathcal{D}}(\boldsymbol{\theta}) \leq \max_{\boldsymbol{\theta}' \in \mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M})} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}') + \mathcal{O}\left(C(\mathcal{M})\rho^2 + \sqrt{\frac{\frac{d}{\rho^2} + d + \log\frac{n}{\delta}}{n-1}}\right) \tag{2}$$

in which $C(\mathcal{M})$ is a constant depends on $\mathcal{M}$.

Remark 1. *The constant $C(\mathcal{M})$ depends on the manifold structure and may scale with dimensions for general manifolds. Additionally, for retractions on the Stiefel manifold used in our practical algorithm and our experiments, the constant can be computed explicitly and independently with (d, k). Specifically, when using the QR factorization or the polar decomposition as the retraction, we have $C(\mathcal{M}) = 1 + \sqrt{2}/2$. Hence, we arrive at the same upper bound as the first inequality in this case.*

The first inequality expresses the generalization ability regarding neighborhood-wise training loss on the tangent space instead of the whole ambient space as in Foret et al. (2021b). Since the tangent space locally reflects the geometry of the manifold around a sufficiently small neighborhood, it leads us to the second inequality, which captures the generalization ability within the manifold. We note that $\mathcal{M}$ can have much smaller intrinsic dimensionality than the ambient space dimension, $d \ll k$. Therefore, the theorem gives us a tighter bound of $\mathcal{O}(d^{1/2})$ compared to $\mathcal{O}(k^{1/2})$ of SAM (Foret et al., 2021b). As a follow-up of this theoretical analysis, we also empirically demonstrate in Section 5.3.3 and Appendix 4c that our method did find the lower-sharpness region compared to prior works.

3.3 Algorithm

From the theorems above, we are thus motivated to find the local minimum in regions with small sharpness. Hence, motivated by the theorems, we propose to define the sharpness on manifolds as:

$$\max_{\boldsymbol{\theta}' \in \mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M})} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}') - \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta})$$

Inspired by the term above, we propose to select the parameter values by solving the sharpness minimization problem:

$$\min_{\boldsymbol{\theta} \in \mathcal{M}} \mathcal{L}^{RSAM}_{\mathcal{S}}(\boldsymbol{\theta}), \text{ where } \mathcal{L}^{RSAM}_{\mathcal{S}}(\boldsymbol{\theta}) = \max_{\boldsymbol{\theta}' \in \mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M})} (\boldsymbol{\theta}')$$

Minimizing the equation above is equivalent to:

$$\begin{aligned}
\min_{\boldsymbol{\theta} \in \mathcal{M}} \mathcal{L}^{RSAM}_{\mathcal{S}}(\boldsymbol{\theta}) &= \min_{\boldsymbol{\theta} \in \mathcal{M}} \max_{\boldsymbol{\theta}' \in \mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M})} (\boldsymbol{\theta}') = \min_{\boldsymbol{\theta} \in \mathcal{M}} \max_{\boldsymbol{\theta}' \in R_{\boldsymbol{\theta}}(\mathcal{B}^o_{\boldsymbol{\theta}}(\rho;\mathcal{T}))} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}') \\
&= \min_{\boldsymbol{\theta} \in \mathcal{M}} \left[\mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}) + \left(\max_{\boldsymbol{\epsilon} \in \mathcal{B}^o_{\boldsymbol{\theta}}(\rho;\mathcal{T})} \mathcal{L}_{\mathcal{S}}(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon})) - \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta})\right)\right] \\
&= \min_{\boldsymbol{\theta} \in \mathcal{M}} \left[\mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}) + \left(\max_{\boldsymbol{\epsilon} \in \mathcal{B}^o_{\boldsymbol{\theta}}(\rho;\mathcal{T})} \langle \text{grad}_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}), \boldsymbol{\epsilon} \rangle_{\boldsymbol{\theta}} + \mathcal{O}(||\boldsymbol{\epsilon}||^2_{\boldsymbol{\theta}})\right)\right] \\
&\approx \min_{\boldsymbol{\theta} \in \mathcal{M}} \left(\mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}) + \max_{\boldsymbol{\epsilon} \in \mathcal{B}^o_{\boldsymbol{\theta}}(\rho;\mathcal{T})} \langle \text{grad}_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}), \boldsymbol{\epsilon} \rangle_{\boldsymbol{\theta}}\right)
\end{aligned}$$

Where the third equality comes from the Taylor expansion on Riemannian manifold in Boumal (2023) and $\text{grad}_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta})$ indicates the Riemannian gradient. Recall the definition that $\mathcal{B}^o_{\boldsymbol{\theta}}(\rho; \mathcal{M}) = \{\boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M} : \|\boldsymbol{\epsilon}\|_2 \leq \rho\}$. We also define the Riemannian metric $\langle \boldsymbol{\epsilon}, \boldsymbol{\epsilon}' \rangle_{\boldsymbol{\theta}} = \boldsymbol{\epsilon}^\top \mathbf{D}_{\boldsymbol{\theta}} \boldsymbol{\epsilon}'$ for some matrix $\mathbf{D}_{\boldsymbol{\theta}}$ that reflects the local geometry at $\boldsymbol{\theta}$, the minimization problem is thus equivalent to:

$$\min_{\boldsymbol{\theta} \in \mathcal{M}} \max_{\|\boldsymbol{\epsilon}\|_2 \leq \rho, \boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}} \left\langle \text{grad}_{\boldsymbol{\theta}}(\mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta})), \boldsymbol{\epsilon} \right\rangle_{\boldsymbol{\theta}} \tag{3}$$

We first attempt to solve the inner maximization problem. Indeed, the problem has the following closed-form solution whose proof can be found in Appendix A.1.1.

Proposition 1. *Let* $\text{grad}_\theta \mathcal{L}(\boldsymbol{\theta})^\top \boldsymbol{D}_{\boldsymbol{\theta}} = \boldsymbol{v}_{\boldsymbol{\theta}}^\top$ *and* $(\boldsymbol{u}_{\boldsymbol{\theta},j})$ *be the system of orthonormal vectors of space formed by* $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$*. The closed-form solution to the maximization problem in Eq. (3) is given by:*

$$\boldsymbol{\epsilon}^* = \rho \sum_j \frac{\text{grad}_\theta \mathcal{L}(\boldsymbol{\theta})^\top \boldsymbol{D}_{\boldsymbol{\theta}} \boldsymbol{u}_{\boldsymbol{\theta},j}}{\sqrt{\sum_j \left[\text{grad}_\theta \mathcal{L}(\boldsymbol{\theta})^\top \boldsymbol{D}_{\boldsymbol{\theta}} \boldsymbol{u}_{\boldsymbol{\theta},j}\right]^2}} \boldsymbol{u}_{\boldsymbol{\theta},j}$$

However, this closed-form solution is impractical in a wide range of cases. Firstly, due to the nested loop computation, the complexity scales poorly with respect to the dimensionality of $\mathcal{M}$. Moreover, finding the set of orthogonal vectors $(\mathbf{u}_{\boldsymbol{\theta},j})$ for a general manifold is not always straightforward in practice. Thus, we propose a more practical yet effective algorithm that first aims to find the solution $\bar{\boldsymbol{\epsilon}}$ to the following relaxed problem:

$$\max_{\|\boldsymbol{\epsilon}\|_2 \leq \rho} \text{grad}_{\boldsymbol{\theta}} (\mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}} \boldsymbol{\epsilon} \tag{4}$$

and then project the solution onto the tangent space $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$ to get $\boldsymbol{\epsilon}^* = \text{Proj}_{\boldsymbol{\theta}}(\bar{\boldsymbol{\epsilon}})$, which gives us an approximated solution to the maximization problem. Indeed, Eq. (4) yields the following solution, whose proof can be found in Appendix A.1.1.

Proposition 2. *The solution to the maximization problem in Eq. (4) is given by*

$$\bar{\boldsymbol{\epsilon}} = \rho \frac{grad_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \boldsymbol{D}_{\boldsymbol{\theta}}}{\|grad_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \boldsymbol{D}_{\boldsymbol{\theta}}\|_2}$$

After finding $\bar{\boldsymbol{\epsilon}}$, we project the solution onto the tangent space and derive the approximated solution $\boldsymbol{\epsilon}^* = \text{Proj}_{\boldsymbol{\theta}}(\bar{\boldsymbol{\epsilon}})$ to the maximization problem in Eq. 3. We will use this approximated solution for RSAM throughout this work, showing that it remarkably improves generalization ability in practice. Moreover, we empirically demonstrate in Section 5.3.1 that compared to the previous exact computation, this approach is notably more efficient and yet remains the same performance. Also, this approximated approach is much more flexible and applicable to a broad category of manifolds since the computation does not involve the orthogonal vectors of the manifolds. One may also notice that we use a matrix $\mathbf{D}_{\boldsymbol{\theta}}$ that can be adapted to learn the local metric at $\boldsymbol{\theta}$. The choice of this matrix is flexible. It can be either $\mathbf{D}_{\boldsymbol{\theta}} = \text{diag}(|\boldsymbol{\theta}_1|, |\boldsymbol{\theta}_2|, \cdots, |\boldsymbol{\theta}_k|)$, or $\mathbf{D}_{\boldsymbol{\theta}} = \mathbf{I}$. In our empirical studies, we use the former and refer to Section 5.3.2 for comparisons between these choices. Then, we solve the outer minimization problem with Riemannian gradient descent. In short, we summarize our algorithm Riemannian Sharpness-Aware Minimization as per Algorithm 1.

Algorithm 1 Riemannian Sharpness-aware Minimization (RSAM)

Input Riemannian manifold $\mathcal{M}$, training set $\mathcal{S} \doteq \cup_{i=1}^n \{(\mathbf{x}_i, \mathbf{y}_i)\}$. Loss function $\ell : \mathcal{W} \times \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}^+$, batch size b, learning rate $\eta > 0$, neighborhood size $\rho > 0$.
Output: Model trained with SAM on manifolds
Initialize weight θ_0 on the manifold $\mathcal{M}$, $t = 0$
while *not converge* **do**
  Sample mini batch $\mathcal{B} = \{(\mathbf{x}_1, \mathbf{y}_1), \cdots, (\mathbf{x}_b, \mathbf{y}_b)\}$
  Compute the batch Riemannian gradient $\text{grad}_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{B}}(\theta) = \text{Proj}_{\boldsymbol{\theta}}\big(\nabla \mathcal{L}_{\mathcal{B}}(\boldsymbol{\theta})\big)$
  Compute $\bar{\boldsymbol{\epsilon}} = \rho \frac{\left(\text{grad}_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{B}}(\boldsymbol{\theta})\right)^\top \mathbf{D}_{\boldsymbol{\theta}}}{\left\| \left(\text{grad}_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{B}}(\boldsymbol{\theta})\right)^\top \mathbf{D}_{\boldsymbol{\theta}} \right\|_2}$, and $\boldsymbol{\epsilon}^* = \text{Proj}_{\boldsymbol{\theta}}(\bar{\boldsymbol{\epsilon}})$
  Ascend step: Compute $\boldsymbol{\theta}^* = R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon}^*)$
  Descend step: Update $\boldsymbol{\theta}_{t+1} = R_{\boldsymbol{\theta}_t}\big(-\eta \text{grad}_{\boldsymbol{\theta}}\big(\mathcal{L}_{\mathcal{B}}(\boldsymbol{\theta}^*)\big)\big)$
end while

4 Practical Methods for Self-Supervised and Supervised Learning

In this section, we discuss the practical applications of RSAM for two settings: *supervised learning* and *self-supervised learning*. We will demonstrate these applications empirically in the next section. Throughout this paper, we are specifically interested in the Stiefel manifolds, that is defined as:

Definition 1 (The Stiefel Manifolds)**.** *The set of $(n \times p)-$dimensional matrices, $p \leq n$, with orthogonal columns and Frobenius inner products forms a Riemannian manifold is called the Stiefel manifold $St(p, n)$*

$$St(p, n) \doteq \{\boldsymbol{X} \in \mathbb{R}^{n \times p} : \boldsymbol{X}^\top \boldsymbol{X} = \boldsymbol{I}_p\}$$

Absil et al. (2008b) proposed multiple retractions for Stiefel manifolds. For the sake of computational complexity, we suggest using the retraction: $R_{\mathbf{X}}(\varepsilon) = \text{qf}(\mathbf{X} + \varepsilon)$ in which $\text{qf}(\mathbf{A})$ denote the $\mathbf{Q}$ factor of the decomposition of $\mathbf{A} \in \mathbb{R}_*^{n \times p}$ as $\mathbf{A} = \mathbf{QR}$. The projection can also be derived as $\text{Proj}_{\mathbf{X}}(\mathbf{v}) = \mathbf{v} - \mathbf{X}\text{Sym}(\mathbf{X}^\top \mathbf{v})$ in which $\text{Sym}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} + \mathbf{A}^\top)$. In this paper, we demonstrate the performance of the Stiefel manifold in two applications: imposing orthogonal convolutional filters in CNN and metric learning for supervised contrastive learning.

4.1 Metric Learning for Self-supervised Learning

In this section, we particularly consider the Supervised Contrastive (SupCon) methodology as proposed by Khosla et al. (2021). For a set of N randomly sampled sample/label pairs, $\{\mathbf{x}_k, \mathbf{y}_k\}_{k=1\cdots N}$, the corresponding batch used for training consists of $2N$ pairs, $\{\tilde{\mathbf{x}}_l, \tilde{\mathbf{y}}_l\}_{l=1,\cdots,2N}$, where $\tilde{\mathbf{x}}_{2k}$ and $\tilde{\mathbf{x}}_{2k-1}$ are random augmentations of $\mathbf{x}_k$, and $\tilde{\mathbf{y}}_{2k-1} = \tilde{\mathbf{y}}_{2k} = \mathbf{y}_k$. We refer to a set of N samples as a "batch" and the set of $2N$ samples as a "multiview batch". Within a multiviewed batch, let $i \in I = \{1, \cdots, 2N\}$ be the index of an arbitrary augmented sample, and let $j(i)$ be the index of the other augmented sample originating from the same source sample. The architecture of SupCon involves two components: **1)** The backbone Encoders, which we denote as $\text{Enc}(\cdot)$; and **2)** The projection head $\text{Proj}(\cdot)$, which is either a linear or fully-connected low-dimensional layer. For any l, we denote $\mathbf{z}_l = \text{Proj}(\text{Enc}(\tilde{\mathbf{x}}_l))$. We also define $A(i) = I/\{i\}$. As proposed by Khosla et al. (2021), $\mathbf{z}_l$'s are then trained with the SupCon objective:

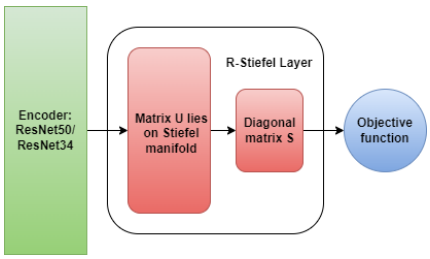


Figure 2: Metric learning with R-Stiefel layer. The projectional layer, which is typically a linear layer or an MLP, is replaced with the R-Stiefel layer consisting of a matrix $\mathbf{U} \in \text{St}(n, p)$ and a diagonal matrix $\mathbf{S}$.

$$\mathcal{L}_{out}^{sup} = \sum_{i \in I} \mathcal{L}_{out,i}^{sup} = \sum_{i \in I} \frac{-1}{|P(i)|} \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_p / \tau)}{\sum_{a \in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)}$$

$$= \mathcal{L}(\mathbf{z}_1 \cdots, \mathbf{z}_{2N}) = \mathcal{L}(\text{Proj}(f(\tilde{\mathbf{x}}_1)) \cdots, \text{Proj}(f(\tilde{\mathbf{x}}_{2N})))$$

Here, $P(i) = \{p \in A(i) : \tilde{\mathbf{y}}_p = \tilde{\mathbf{y}}_i\}$. Instead of using the Euclidean dot product, we replace it with the Mahalanobis distant $\langle \cdot, \cdot \rangle$ in which $\langle \mathbf{h}, \mathbf{h}' \rangle = \mathbf{h}^\top \mathbf{M} \mathbf{h}'$, and $\mathbf{M}$ is learnable. By doing so, $\mathbf{M}$ can be learned to take into account the local geometry of the parameter space, and the neighborhood becomes an adaptive ellipsoid instead of an open ball that treats every dimension identically. Singular Value Decomposition yields $\mathbf{M} = \mathbf{UDU}^\top = \mathbf{UD}^{1/2}\mathbf{D}^{1/2}\mathbf{U}^\top$. Denote $\mathbf{S} = \mathbf{D}^{1/2}$, it follows that:

$$\langle \mathbf{h}, \mathbf{h}' \rangle = \mathbf{h}^\top \mathbf{M} \mathbf{h}' = (\mathbf{hUS})^\top \cdot (\mathbf{h}'\mathbf{US})$$

Thus, instead of optimizing $\mathcal{L}(\text{Proj}(\text{Enc}(\tilde{\mathbf{x}}_1)), \cdots, \text{Proj}(\text{Enc}(\tilde{\mathbf{x}}_{2N})))$, we will optimize $\mathcal{L}(\text{Proj}(\text{Enc}(\tilde{\mathbf{x}}_1))\mathbf{US}, \cdots, \text{Proj}(\text{Enc}(\tilde{\mathbf{x}}_{2N}))\mathbf{US})$ in which $\mathbf{U}$ is a rotational matrix on the Stiefel manifold, and $\mathbf{S}$ is a diagonal matrix. From now on, we will call the layer that multiplies with the matrix $\mathbf{US}$ an R-Stiefel layer. We refer to Figure 2 for illustration. Such modification can be done not only on the SupCon loss function but also on different loss functions involving distance calculations such as triplet loss (Roy et al., 2019). Since $\mathbf{U}$ is constrained to lies on the Stiefel manifold, we will optimize it with RSAM, and the rest of the parameters, including the backbone and the diagonal matrix $\mathbf{S}$, will be learned via traditional optimizers such as SAM or SGD.

4.2 Orthogonal Convolutional Neural Network

Orthogonality of convolutional filters has been proven to be useful for several purposes, such as alleviating gradient vanishing or exploding phenomenon (Xie et al., 2017), or decorrelating the filter banks so that they learn distinct features (Wang et al., 2020). For each ℓ, let $\{\mathbf{W}_i\}_{i=1}^C$ be the set of

convolutional kernels in $\ell-$th layer, in which $\mathbf{W}_i \in \mathbb{R}^{WHM}$. Previous works impose orthogonality by introducing orthogonal regularizers such as $\mathcal{L}_{\text{ortho}} = \frac{\lambda}{2}\sum_{i=1}^{D} \|\mathbf{W}_i^\top \mathbf{W} - \mathbf{I}\|_2^2$ (Xie et al., 2017), or a self-convolution regularization term of the kernels (Wang et al., 2020) to encourage orthogonality between the convolutional kernels. In this section, we propose eliminating those regularizers and directly enforcing the kernels to be always orthogonal during training. Indeed, we flatten the kernels $\mathbf{W}_i$ into column vectors of shape $W \times H \times M$. Let $\mathbf{W}_\ell$ be the matrix with the columns formed by $\mathbf{W}_i's$. The advantage of RSAM is that we can guarantee $\mathbf{W}_\ell$ always lies on the Stiefel manifold $\text{St}(W \times H \times M, C)$ during training, which means that:

$$\mathbf{W}_\ell^\top \mathbf{W}_\ell = \mathbf{I}_d$$

always holds throughout training, therefore guarantees orthonormality between the kernels on that specific layer ℓ. By directly guaranteeing the parameters to reside within its true geometry, it is expected to improve the robustness of the learned model. In the next section, we will demonstrate that simply imposing orthogonality onto the first two convolutional layers of the architecture by training with RSAM can yield a notable generalization ability improvement.

5 Experiments

5.1 Training details

To assess RSAM's efficacy, we experiment on various vision datasets (including CIFAR10, CIFAR100, and FGVCAircraft) and different architectures (including ResNet50 and ResNet34). We conduct two sets of experiments: the standard supervised classification from scratch and supervised contrastive learning. All the experiments were trained for 500 epochs. The learning rates η of SGD, SAM, and RSAM are set to 0.1 with a cosine learning rate scheduler throughout the experiments. ρ in SAM is set to 0.1, and ρ in RSAM is set to 0.5. We trained our model with a batch size of 256 on CIFAR100 and CIFAR10. We specifically note that the FGVCAircraft dataset has a higher resolution, so we use a smaller batch size of 64 on this dataset for all methodologies.

5.2 Experimental results

Supervised Learning. We examine the classification accuracy with cross-entropy loss for the first set of experiments, optimizing by SGD, Sharpness Aware Minimization (SAM), and our algorithm RSAM. In this scenario, RSAM is used for imposing orthogonality of convolutional layers. Specifically, we imposed orthogonality on the first two convolutional layers of the architecture in all settings. Table 1 shows that RSAM generalizes better than the baselines in this standard training setting, with an improvement of 1-3% compared to SGD and more than 1% higher than SAM.

Method	**CIFAR100**		**CIFAR10**		**FGVCAircraft**	
	ResNet50	ResNet34	ResNet50	ResNet34	ResNet50	ResNet34
CE + SGD	74.62	73.67	94.56	95.14	82.44	78.79
CE + SAM	75.04	75.05	95.39	95.52	83.01	80.56
CE + RSAM	**77.78**	**76.36**	**96.32**	**96.10**	**84.68**	**83.12**

Table 1: Top-1 classification accuracy for supervised learning settings. We compare cross-entropy training with SGD with momentum, SAM, and RSAM (Ours). RSAM is used to impose orthogonality convolutional of the filter banks.

Self-Supervised Learning. The second set of experiments has two stages. SupCon is trained with SGD with momentum, SAM, and RSAM in pretraining. Then, in the second stage, we conduct linear evaluation, that is, to freeze the parameters and train a linear classifier. We note that in the pretrained step, the projectional layer of SGD and SAM are linear layers, while RSAM's is the R-Stiefel layer as discussed in Section 4.1. Therefore, the applications of RSAM in this setting are two-fold: RSAM is used to impose orthogonality on the convolutional layers and used for the R-Stiefel during pretraining. As shown in Table 2, RSAM consistently outperforms the baselines. Furthermore,

we note that on ResNet50, RSAM made a remarkable accuracy of 81.62% on CIFAR100, which outperforms 5% compared to SupCon on the same setting.

We also further note that RSAM involves additional manifold computations, which are expected to be slower. However, as we will show in Appendix. A.2.1, such difference in runtime is negligible and, therefore, worth the trade-off for better performance. Furthermore, we show in Appendix A.2.2 that RSAM has successfully found the low-sharpness region as suggested by our theoretical analysis.

Method	**CIFAR100**		**CIFAR10**		**FGVCAircraft**	
	ResNet50	ResNet34	ResNet50	ResNet34	ResNet50	ResNet34
SupCon + SGD	75.29	74.04	95.99	95.34	82.03	78.19
SupCon + SAM	76.73	76.91	96.31	96.07	82.84	81.73
SupCon + RSAM	**81.62**	**80.51**	**96.86**	**96.65**	**84.73**	**84.52**

Table 2: Top-1 classification accuracy for self-supervised learning settings with SupCon loss. SupCon is trained with SGD with momentum, SAM, and RSAM (Ours). RSAM is used for metric learning and imposing orthogonality on the convolutional operations.

5.3 Ablation Studies

5.3.1 Approximated solution vs. Closed-form solution Comparision

In this section, we justify the usage of the approximated solution as per Eq. (2) instead of the exact solution as derived in Eq. (1) by contrasting the two approaches. It is noteworthy that the exact solution contains the orthogonal vectors $(\mathbf{u}_{\boldsymbol{\theta},j})$. For each matrix $\mathbf{X}$ on the Stiefel manifold, these vectors are given as the set of matrices $\{\mathbf{XS}\}$, where $\mathbf{S}$ is any p-by-p matrices. However, for a general manifold, the computation of these orthogonal vectors is not always available. Thus, the approximated approach of RSAM gives us more flexibility and applies to other manifolds beyond the Stiefel manifolds. Moreover, the exact solution in Eq. (1) contains nested loops that sum over the orthogonal vectors. Therefore, it is expected to be $\mathcal{O}(p^2)$ asymptotically slower than RSAM. We refer to Table 3 for an empirical comparison, showing that both approaches have roughly the same accuracy. In contrast, the exact solution is about 1.75x times slower. Furthermore, when we imposed orthogonality on convolutional layers with more than 512 kernels, the exact solution failed to complete an epoch within a reasonable time. Thus, using the approximated solution as we did in RSAM is preferable.

Method	**CIFAR100**		**CIFAR10**		**FGVCAircraft**	
	Accuracy	Runtime	Accuracy	Runtime	Accuracy	Runtime
RSAM (exact)	75.21	$78.5_{\pm 1.2}$	**96.15**	$78.9_{\pm 1.3}$	**83.31**	$154.5_{\pm 2.6}$
RSAM (approx)	**76.36**	$\mathbf{52.6}_{\pm 1.7}$	96.10	$\mathbf{51.1}_{\pm 1.9}$	83.12	$\mathbf{133.0}_{\pm 1.4}$

Table 3: Comparision between the exact solution and approximated solution to Eq. (3) in terms of Top-1 classification accuracy and per-epoch wallclock runtime. The experiment is conducted on ResNet34.

5.3.2 Choices of $\mathbf{D}_\theta$ Comparison

We recall that the procedure of RSAM in 1 involves the matrix $\mathbf{D}_{\boldsymbol{\theta}}$, which serves to adjust the metric on the Euclidean tangent space. In Table 4, we compare the performance on the standard training settings for different choices of $\mathbf{D}_{\boldsymbol{\theta}}$. As shown in the table, an adaptive choice for $\mathbf{D}_{\boldsymbol{\theta}}$ does have an effect on the final performance, but this effect is negligible, and hence RSAM's performance is not highly dependent on the choice for this $\mathbf{D}_{\boldsymbol{\theta}}$.

5.3.3 SAM vs. RSAM: behavioral comparison

In this ablation, we design a simple experiment on the dataset MNIST to show a particular case where RSAM is favorably robust. In particular, we train a simple PCA-style autoencoder that aims

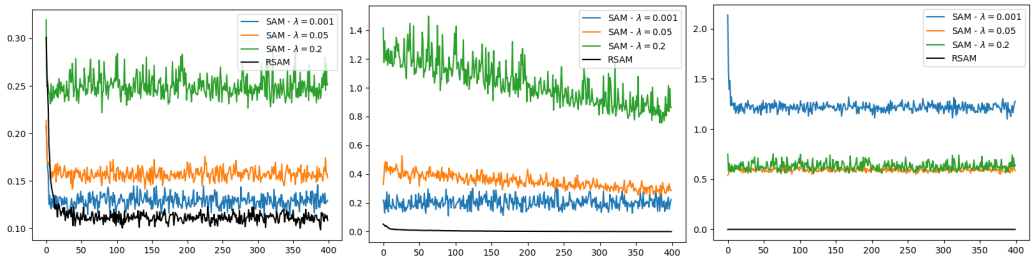
Method	**CIFAR100**		**CIFAR10**	
	ResNet34	ResNet50	ResNet34	ResNet50
$\mathbf{D}_{\boldsymbol{\theta}} = \text{diag}(\lvert\boldsymbol{\theta}_1\rvert, \lvert\boldsymbol{\theta}_2\rvert, \cdots, \lvert\boldsymbol{\theta}_k\rvert)$	**76.36**	**77.78**	96.10	**96.32**
$\mathbf{D}_{\boldsymbol{\theta}} = \mathbf{I}$	76.02	76.91	**96.21**	96.21

Table 4: Top-1 Accuracy on standard training settings for different choices of $\mathbf{D}_{\boldsymbol{\theta}}$.

to find an orthogonal matrix $\mathbf{W}$ that encodes each input $\mathbf{x}$ into lower-dimensional $\mathbf{z} = \mathbf{x}\mathbf{W}$, and then decodes as $\tilde{\mathbf{x}} = \mathbf{z}\mathbf{W}^\top$. The encoded vector $\mathbf{z}$ is then used for the classification task. Therefore, the objective that we will minimize is the reconstruction loss with a classification loss act as a regularizer. Since $\mathbf{W}$ is constrained to be orthogonal, it lies on a Stiefel manifold. To enforce orthogonality with SAM, we need to add an *orthogonal regulazrizer* $\|\mathbf{W}^\top\mathbf{W} - \mathbf{I}_d\|_2^2$. Thus, our objective function is:

$$\mathcal{L}_{\mathcal{S}}(\mathbf{W}) = \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{x}_i - \tilde{\mathbf{x}}_i\|_2^2 + \beta\text{CrossEntropyLoss}(\mathbf{z}_i, \mathbf{y}_i) + \lambda\|\mathbf{W}^\top\mathbf{W} - \mathbf{I}_d\|_2^2$$

In this set of experiments, we set the batch size to 16, the learning rate to 0.1, $\beta = 0.1$, and $\rho = 0.3$. In Figure 3, we report **1)** the loss value over time, **2)** The sharpness of the loss function over time, and **3)** the values of the orthogonal regularize, which measures how orthogonal the parameters were. Indeed, in terms of loss function convergence, the smaller λ is, the better SAM can keep up with RSAM because the orthogonal regularization has less impact on the final loss function. However, for smaller values λ, $\mathbf{W}$ fails to be orthogonal, demonstrating that SAM is remarkably sensitive to this orthogonal regularization. Hence, we emphasize that by directly enforcing $\mathbf{W}$ to be on the Stiefel manifold, RSAM eliminates this vulnerability, remarkably reducing the sharpness and leading to better loss convergence. In short, the ablation suggests that in certain scenarios, taking into account the intrinsic geometry of the parameters can notably enhance the model's robustness and performance, and we propose that RSAM has successfully done so.



(a) Loss values **(b)** Sharpness **(c)** Values of $\|\mathbf{W}^\top\mathbf{W} - \mathbf{I}_d\|_2^2$

Figure 3: Comparision between SAM with different λ's and RSAM (black) over 50 epochs

6 Conclusion

In this work, we have generalized the SAM technique by introducing a novel notion of sharpness on Riemannian manifolds. On the theoretical side, we backed this notion with a theorem that characterizes the generalization ability in terms of neighborhood-wise training loss on the manifolds, which demonstrated a tighter bound compared to existing works. Motivated by this theoretical analysis, we propose RSAM, which considers the parameter space's intrinsic geometry and seeks regions with flat surfaces on Riemannian manifolds. On the experimental side, the effectiveness of RSAM is demonstrated on different tasks with various datasets and models, in which RSAM outperforms the comparative methodologies by a notable margin.

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-Antoine Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization Algorithms on Matrix Manifolds*, volume 78. 12 2008a. ISBN 978-0-691-13298-3. doi: 10.1515/9781400830244.

Pierre-Antoine Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization Algorithms on Matrix Manifolds*, volume 78. 12 2008b. ISBN 978-0-691-13298-3. doi: 10.1515/9781400830244.

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.

Silvere Bonnabel. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229, sep 2013. doi: 10.1109/tac.2013.2254619. URL https://doi.org/10.1109%2Ftac.2013.2254619.

Nicolas Boumal. *An introduction to optimization on smooth manifolds*. Cambridge University Press, 2023. doi: 10.1017/9781009166164. URL https://www.nicolasboumal.net/book.

Nicolas Boumal and P.-A. Absil. Low-rank matrix completion via preconditioned optimization on the grassmann manifold. *Linear Algebra and its Applications*, 475:200–239, 2015. ISSN 0024-3795. doi: https://doi.org/10.1016/j.laa.2015.02.027. URL https://www.sciencedirect.com/science/article/pii/S0024379515001342.

Nicolas Boumal, P-A Absil, and Coralia Cartis. Global rates of convergence for nonconvex optimization on manifolds. *IMA Journal of Numerical Analysis*, 39(1):1–33, feb 2018. doi: 10.1093/imanum/drx080. URL https://doi.org/10.1093%2Fimanum%2Fdrx080.

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.

Wei Dai, Ely Kerman, and Olgica Milenkovic. A geometric approach to low-rank matrix completion. *IEEE Transactions on Information Theory*, 58(1):237–247, 2012. doi: 10.1109/TIT.2011.2171521.

Jiawei Du, Daquan Zhou, Jiashi Feng, Vincent YF Tan, and Joey Tianyi Zhou. Sharpness-aware training for free. *arXiv preprint arXiv:2205.14083*, 2022.

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

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021b. URL https://openreview.net/forum?id=6Tm1mposlrM.

Zhi Gao, Yuwei Wu, Yunde Jia, and Mehrtash Harandi. Learning to optimize on spd manifolds. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7697–7706, 2020. doi: 10.1109/CVPR42600.2020.00772.

Keith Hall and Thomas Hofmann. Learning curved multinomial subfamilies for natural language processing and information retrieval. pp. 351–358, 01 2000.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

Jean Kaddour, Linqing Liu, Ricardo Silva, and Matt Kusner. When do flat minima optimizers work? In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022a. URL https://openreview.net/forum?id=vDeh2yxTvuh.

Jean Kaddour, Linqing Liu, Ricardo Silva, and Matt J. Kusner. Questions for flat-minima optimization of modern neural networks. *CoRR*, abs/2202.00661, 2022b. URL https://arxiv.org/abs/2202.00661.

Hiroyuki Kasai, Pratik Jawanpuria, and Bamdev Mishra. Riemannian adaptive stochastic gradient algorithms on matrix manifolds. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3262–3271. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/kasai19a.html.

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. *CoRR*, abs/1609.04836, 2016. URL http://arxiv.org/abs/1609.04836.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2021.

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.

Zhouzi Li, Zixuan Wang, and Jian Li. Analyzing sharpness along gd trajectory: Progressive sharpening and edge of stability, 2022.

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.

David Luenberger. The gradient projection method along geodesics. *Management Science*, 18: 620–631, 07 1972. doi: 10.1287/mnsc.18.11.620.

David A. McAllester. Pac-bayesian model averaging. In *Annual Conference Computational Learning Theory*, 1999. URL https://api.semanticscholar.org/CorpusID:11948100.

Thomas Möllenhoff and Mohammad Emtiyaz Khan. SAM as an optimal relaxation of bayes. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=k4fevFqSQcX.

Van-Anh Nguyen, Trung Le, Anh Tuan Bui, Thanh-Toan Do, and Dinh Phung. Optimal transport model distributional robustness. In *Advances in Neural Information Processing Systems*, 2023a.

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

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.

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.

Soumava Roy, Mehrtash Harandi, Richard Nock, and Richard Hartley. Siamese networks: The tale of two manifolds. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3046–3055, 2019. doi: 10.1109/ICCV.2019.00314.

Hiroyuki Sato, Hiroyuki Kasai, and Bamdev Mishra. Riemannian stochastic variance reduced gradient algorithm with retraction and vector transport. *SIAM Journal on Optimization*, 29(2): 1444–1472, jan 2019. doi: 10.1137/17m1116787. URL https://doi.org/10.1137%2F17m1116787.

Jiayun Wang, Yubei Chen, Rudrasis Chakraborty, and Stella X. Yu. Orthogonal convolutional neural networks, 2020.

Di Xie, Jiang Xiong, and Shiliang Pu. All you need is beyond a good init: Exploring better solution for training extremely deep convolutional neural networks with orthonormality and modulation, 2017.

Hongyi Zhang, Sashank J. Reddi, and Suvrit Sra. Riemannian svrg: Fast stochastic optimization on riemannian manifolds, 2017.

Juntang Zhuang, Boqing Gong, Liangzhe Yuan, Yin Cui, Hartwig Adam, Nicha Dvornek, Sekhar Tatikonda, James Duncan, and Ting Liu. Surrogate gap minimization improves sharpness-aware training. *arXiv preprint arXiv:2203.08065*, 2022.

A APPENDIX

A.1 ALL PROOFS

A.1.1 PROOF OF PROPOSITION 1

Firstly, we restate the optimization problem:

$$\max_{\|\boldsymbol{\epsilon}\|_2^2 \leq \rho^2, \boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}} \left\langle \text{grad}_{\boldsymbol{\theta}}(\mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta})), \boldsymbol{\epsilon} \right\rangle_{\theta} \tag{5}$$

Let $\text{grad}_{\theta}\mathcal{L}(\boldsymbol{\theta})^{\top}\mathbf{D}_{\boldsymbol{\theta}} = \mathbf{v}_{\boldsymbol{\theta}}^{\top}$ and $(\mathbf{u}_{\boldsymbol{\theta},j})$ be an orthonormal basis of $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$. Then, it follows that:

$$\mathbf{u}_{\boldsymbol{\theta},i}^{\top}\mathbf{u}_{\boldsymbol{\theta},j} = \delta_{i,j}.$$

Under the assumption that the $\mathbf{u}_{\boldsymbol{\theta},j}$ form a basis in the tangent space at $\boldsymbol{\theta}$, there exist β_j such that

$$\boldsymbol{\epsilon} = \sum_j \beta_j \mathbf{v}_{\boldsymbol{\theta},j}.$$

It deduces that

$$\boldsymbol{\epsilon}^{\top}\boldsymbol{\epsilon} = \Big[\sum_j \beta_j \mathbf{u}_{\boldsymbol{\theta},j}\Big]^{\top}\Big[\sum_j \beta_j \mathbf{u}_{\boldsymbol{\theta},j}\Big] = \sum_j \beta_j^2.$$

We have the Lagrangian objective being:

$$\mathbf{v}_{\boldsymbol{\theta}}^{\top}\boldsymbol{\epsilon} + \lambda\big[\boldsymbol{\epsilon}^{\top}\boldsymbol{\epsilon} - \rho^2\big] = \sum_j \beta_j \mathbf{v}_{\boldsymbol{\theta}}^{\top}\mathbf{u}_{\boldsymbol{\theta},j} + \lambda\Big[\sum_j \beta_j^2 - \rho^2\Big].$$

Taking derivative with respect to λ and β_j, we get the following system of equations:

$$\sum_j \beta_j^2 = \rho^2$$

$$\mathbf{v}_{\boldsymbol{\theta}}^{\top}\mathbf{u}_{\boldsymbol{\theta},j} + 2\lambda\beta_j = 0.$$

Solving the second equation of the system yields:

$$\beta_j = -\frac{1}{2\lambda}\mathbf{v}_{\boldsymbol{\theta}}^{\top}\mathbf{u}_{\boldsymbol{\theta},j}$$

Substituting into the first equation of the system, we get:

$$\frac{1}{4\lambda^2}\sum_j \big[\mathbf{v}_{\boldsymbol{\theta}}^{\top}\mathbf{u}_{\boldsymbol{\theta},j}\big]^2 = \rho^2$$

$$\Rightarrow \frac{1}{2\lambda} = \pm\rho\Big\{\sum_j \big[\mathbf{v}_{\boldsymbol{\theta}}^{\top}\mathbf{u}_{\boldsymbol{\theta},j}\big]^2\Big\}^{-\frac{1}{2}}$$

Then, the optimal solution to the maximization problem is given by:

$$\begin{aligned}\boldsymbol{\epsilon}^* &= \rho\sum_j \frac{\mathbf{v}_{\boldsymbol{\theta}}^{\top}\mathbf{u}_{\boldsymbol{\theta},j}}{\sqrt{\sum_j \big[\mathbf{u}_{\boldsymbol{\theta}}^{\top}\mathbf{u}_{\boldsymbol{\theta},j}\big]^2}}\mathbf{u}_{\boldsymbol{\theta},j} \\ &= \rho\sum_j \frac{\text{grad}_{\theta}\mathcal{L}(\boldsymbol{\theta})^{\top}\mathbf{D}_{\boldsymbol{\theta}}\mathbf{u}_{\boldsymbol{\theta},j}}{\sqrt{\sum_j \big[\text{grad}_{\theta}\mathcal{L}(\boldsymbol{\theta})^{\top}\mathbf{D}_{\boldsymbol{\theta}}\mathbf{u}_{\boldsymbol{\theta},j}\big]^2}}\mathbf{u}_{\boldsymbol{\theta},j}\end{aligned}$$

A.1.2 PROOF OF PROPOSITION 2

Firstly, we restate the optimization problem:

$$\max_{\|\boldsymbol{\epsilon}\|_2 \leq \rho} \text{grad}_{\boldsymbol{\theta}}(\mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}))^{\top}\mathbf{D}_{\boldsymbol{\theta}}\boldsymbol{\epsilon} \tag{6}$$

We prove that the optimal solution of the problem in Eq. (6) occurs on the boundary. Suppose on the contrary that the optimal solution is $\boldsymbol{\epsilon}^*$ satisfies $(\boldsymbol{\epsilon}^*)^\top \boldsymbol{\epsilon}^* = (\rho^*)^2 < \rho^2$. Then, $-\text{grad}_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})^\top \mathbf{D}_{\boldsymbol{\theta}} \boldsymbol{\epsilon}^* \leq 0$, otherwise we may replace $\boldsymbol{\epsilon}^*$ with $-\boldsymbol{\epsilon}^*$ that still satisfies the constraint and arrive at a strictly smaller objective. However, if we replace $\boldsymbol{\epsilon}^*$ with $\overline{\boldsymbol{\epsilon}} = \boldsymbol{\epsilon}^* \sqrt{\frac{\rho}{\rho^*}}$, it follows:

$$\overline{\boldsymbol{\epsilon}}^\top \overline{\boldsymbol{\epsilon}} = \rho^2$$

and arrive at a smaller objective since $\overline{\boldsymbol{\epsilon}} > \boldsymbol{\epsilon}^*$, which is a contradiction since we are assuming that $\boldsymbol{\epsilon}^*$ is the optimal solution.

Therefore, the optimal solution $\boldsymbol{\epsilon}^*$ occurs on the boundary. Thus, the problem is reduced to:

$$\text{maximize: } \text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}} \boldsymbol{\epsilon} \tag{7a}$$

$$\text{subject to: } \boldsymbol{\epsilon}^\top \boldsymbol{\epsilon} - \rho^2 = 0 \tag{7b}$$

This problem has the Lagrangian:

$$L(\boldsymbol{\epsilon}, \lambda) = -\text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}} \boldsymbol{\epsilon} + \lambda(\boldsymbol{\epsilon}^\top \boldsymbol{\epsilon} - \rho^2)$$

The stationary point of L satisfies $\frac{\partial L}{\partial \boldsymbol{\epsilon}} = 0$ and $\frac{\partial L}{\partial \lambda} = 0$, which is equivalent to:

$$\begin{aligned} -\text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}} + 2\lambda \boldsymbol{\epsilon} &= 0 \\ \boldsymbol{\epsilon}^\top \boldsymbol{\epsilon} &= \rho^2 \end{aligned}$$

Thus, we have the system of equations:

$$\begin{aligned} \boldsymbol{\epsilon} &= \frac{1}{2\lambda} \text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}} \\ \boldsymbol{\epsilon}^\top \boldsymbol{\epsilon} &= \rho^2 \end{aligned}$$

Substituting the first equation into the second one, we get:

$$\frac{1}{4\lambda^2} \mathbf{D}_{\boldsymbol{\theta}}^\top \text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta})) \text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}} = \rho^2$$

Which follows that:

$$\frac{1}{2\lambda} = \frac{\rho}{\Big((\text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}})^\top (\text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}}) \Big)^{\frac{1}{2}}} = \frac{\rho}{\|\text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}}\|_2}$$

Therefore, the maximization problem governs a closed-form solution:

$$\overline{\boldsymbol{\epsilon}} = \rho \frac{\text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}}}{\|\text{grad}_{\boldsymbol{\theta}}(\mathcal{L}(\boldsymbol{\theta}))^\top \mathbf{D}_{\boldsymbol{\theta}}\|_2}$$

A.1.3 PROOF OF THEOREM 1

Before going into the proof of our main theorems, we state an additional notation that we will be using throughout the proof. Indeed, for each $\boldsymbol{\theta} \in \mathcal{M}$, we define $A_{\boldsymbol{\theta}} \in \mathbb{R}^{k \times d}$ to be a column-orthogonal matrix whose columns form a basis of the tangent space $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$. We can construct $A_{\boldsymbol{\theta}}$ so that it is coordinate-wise smooth. To do so, notice that $\mathcal{M}$ is a C^∞ embedded submanifold, so if we denote U to be a neighborhood of $\boldsymbol{\theta}$ and a chart $\phi : (x^1, \cdots, x^k) : U \to \mathbb{R}^k$, we can define the ordered basis of $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$ as:

$$\forall i \in \{1, 2, \cdots, k\}, \forall f \in C^\infty(\mathcal{M}) : \frac{\partial}{\partial x^i}\Big|_{\boldsymbol{\theta}}(f) = \Big(\frac{\partial}{\partial x^i}(f \circ \phi^{-1}) \Big)(\phi(\boldsymbol{\theta}))$$

We stacked the basis above into the columns of a matrix and then applied the Gram-Schmidt matrix; we got a smooth matrix $A_{\boldsymbol{\theta}}$ whose columns are orthogonal, that $A_{\boldsymbol{\theta}}^\top A_{\boldsymbol{\theta}} = \mathbf{I}_d$. Recall that by convention, the tangent space $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$ uses the coordinate system with the current $\boldsymbol{\theta}$ as the origin. Hence

$\epsilon \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$ specifies the offset. Also, since the tangent space can locally reflect the geometry of the manifold, for each point $\boldsymbol{\theta} \in \mathcal{M}$, we only consider a sufficiently large and compact neighborhood B of $\boldsymbol{\theta}$ on the tangent space, and that there is a collection of predefined regions R_j such that $B \subset \cup_j R_j$ in which:

$$R_j = \left\{\boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M} : (\boldsymbol{\epsilon} - \overline{\boldsymbol{\epsilon}}_j)^\top (A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)^* (\boldsymbol{\epsilon} - \overline{\boldsymbol{\epsilon}}_j) \leq r_j^2\right\},$$

Here, $\{\overline{\boldsymbol{\epsilon}}_j\}$ is a predefined set of points on $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$. We also define $r = \max_j r_j$, and additionally note that $A_{\boldsymbol{\theta}}^\top A_{\boldsymbol{\theta}} = \mathbf{I}_d$, for all $j = 1, 2, \ldots, J$, and $\left(A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top\right)^*$ denotes the Moore-Penrose inverse of $\left(A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top\right)$.

A.1.4 Proof of First inequality

(The upper-bound on the tangent space $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$): For any small $\rho > 0$ and $\delta \in [0; 1]$, with a high probability $1 - \delta$ over training set $\mathcal{S}$ generated from a distribution $\mathcal{D}$, the following holds:

$$\mathcal{L}_D(\boldsymbol{\theta}) \leq \max_{\boldsymbol{\theta}' \in \mathcal{B}_{\boldsymbol{\theta}}^a(\rho; \mathcal{T})} \mathcal{L}_S(\boldsymbol{\theta}') + \sqrt{\frac{\mathcal{O}(\frac{d}{\rho^2} + d + \log \frac{n}{\delta})}{n-1}}$$

Proof. Firstly, we recall the definition of the offset ball in the tangent space $\mathcal{B}_{\boldsymbol{\theta}}^o(\rho; \mathcal{M}) = \{\boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M} : \|\boldsymbol{\epsilon}\|_2 \leq \rho\}$, and the absolute ball that appears in the theorem statement $\mathcal{B}_{\boldsymbol{\theta}}^a(\rho; \mathcal{M}) = \boldsymbol{\theta} + \mathcal{B}_{\boldsymbol{\theta}}^o(\rho; \mathcal{M}) = \{\boldsymbol{\theta} + \boldsymbol{\epsilon} : \boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M} \text{ and } \|\boldsymbol{\epsilon}\|_2 \leq \rho\}$. Conventionally, the tangent space $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$ uses the coordinate system with $\boldsymbol{\theta}$ being the origin, so for $\boldsymbol{\theta}' = \boldsymbol{\theta} + \boldsymbol{\epsilon}$ on the tangent space, we can write $\mathcal{G}(\boldsymbol{\epsilon}) = \mathcal{L}(\boldsymbol{\theta}')$, which means that $\mathcal{L}(\boldsymbol{\theta}) = \mathcal{G}(0)$, and from now on we will analyze the $d-$dimensional Euclidean tangent space $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$. Accordingly, we will prove that:

$$\mathcal{G}_D(0) \leq \max_{\epsilon \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}, \|\boldsymbol{\epsilon}\| \leq \rho} \mathcal{G}_S(\boldsymbol{\epsilon}) + \sqrt{\frac{\mathcal{O}(\frac{d}{\rho^2} + d + \log \frac{n}{\delta})}{n-1}}$$

Let $\delta > 0$ be a small positive constant. According to the PAC-Bayes generalization bound of McAllester (1999), for any prior distribution $P(\boldsymbol{\theta})$, with probability at least $1 - \delta$ over the choice of training set S, it holds that:

$$\mathbb{E}_{Q(\boldsymbol{\epsilon})}\left[\mathcal{G}_D(\boldsymbol{\epsilon})\right] \leq \mathbb{E}_{Q(\boldsymbol{\epsilon})}\left[\mathcal{G}_S(\boldsymbol{\epsilon})\right] + \sqrt{\frac{\mathsf{KL}\big(Q(\boldsymbol{\epsilon}) || P(\boldsymbol{\epsilon})\big) + \log \frac{n}{\delta}}{2n-2}}, \tag{10}$$

for any posterior distribution $Q(\boldsymbol{\epsilon})$ over the space of $\boldsymbol{\epsilon}$.

Here we consider $Q(\boldsymbol{\epsilon}) = \mathcal{N}(0, \rho^2 A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)$. By imposing such posterior on $\boldsymbol{\epsilon}$, it means that we have $\boldsymbol{\epsilon} = A_{\boldsymbol{\theta}} h$ in which $h \sim \mathcal{N}(0, \rho^2 \mathbf{I}_d)$. To minimize the bound at Eq. 10, we want to choose the prior $P(\boldsymbol{\epsilon})$ that minimizes the KL term, and the prior should be independent of the training set S. Now we considered the predefined collection of points $\{\overline{\boldsymbol{\epsilon}}_j\}_{j=1}^J$ whose neighborhoods covers a sufficiently large neighborhood of $\boldsymbol{\theta}$ on the tangent space $\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$, and a predefined set of J prior distributions $\{P_j(\boldsymbol{\epsilon})\}_{j=1}^J$ in which $P_j(\boldsymbol{\epsilon}) = \mathcal{N}(\overline{\boldsymbol{\epsilon}}_j, \rho^2 A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)$, and we select the closest distribution from this collection. According to the intersection of the training sets for which Eq. 10 holds, we can say that Eq. 10 holds for all $P_j(\boldsymbol{\epsilon})$ over the intersection. By the union bound theorem, the probability over the choice of the intersection is at least $1 - \sum_{j=1}^J \delta_j$. Letting $\delta_j = \delta / J$, we derive that for all $P_j(\boldsymbol{\epsilon})$, with probability at least $1 - \delta$ over the choice of the training set S, the following holds for all $Q(\boldsymbol{\epsilon})$ and $j = 1, \ldots, J$

$$\mathbb{E}_{Q(\boldsymbol{\epsilon})}\left[\mathcal{G}_D(\boldsymbol{\epsilon})\right] \leq \mathbb{E}_{Q(\boldsymbol{\epsilon})}\left[\mathcal{G}_S(\boldsymbol{\epsilon})\right] + \sqrt{\frac{\mathsf{KL}\big(Q(\boldsymbol{\epsilon}) \| P_j(\boldsymbol{\epsilon})\big) + \log \frac{n}{\delta} + \log J}{2n-2}}$$

We choose the prior $P_j(\boldsymbol{\epsilon})$ as close to $Q(\boldsymbol{\epsilon})$. Indeed, the KL divergence term has the following form

$$\begin{aligned}
\mathsf{KL}(Q \| P_j) &= \frac{1}{2}\left[\mathsf{tr}\Big((A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)^* (A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)\Big) + \frac{1}{\rho^2} \overline{\boldsymbol{\epsilon}}_j^\top (A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)^* \overline{\boldsymbol{\epsilon}}_j + \log \frac{|A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top|}{|A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top|} - k\right] \\
&= \frac{1}{2}\left[d - k + \frac{1}{\rho^2} \overline{\boldsymbol{\epsilon}}_j^\top (A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)^* \overline{\boldsymbol{\epsilon}}_j\right]
\end{aligned}$$

Now, we choose j^* such that $0 \in R_{j^*}$. According to our assumption, we have:

$$\overline{\boldsymbol{\epsilon}}_j^\top (A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)^* \overline{\boldsymbol{\epsilon}}_j \leq r_j^2 \leq r^2$$

Therefore, we have the inequality:

$$\mathsf{KL}(Q \| P_j) \leq \frac{1}{2}(d - k + \frac{r^2}{\rho^2}) \leq \frac{d}{2} + \frac{r^2}{2\rho^2}$$

Plugging into the inequality 10, it follows that:

$$\mathbb{E}_{Q(\boldsymbol{\epsilon})}\big[\mathcal{G}_D(\boldsymbol{\epsilon})\big] \leq \mathbb{E}_{Q(\boldsymbol{\epsilon})}\big[\mathcal{G}_S(\boldsymbol{\epsilon})\big] + \sqrt{\frac{\frac{d}{2} + \frac{r^2}{2\rho^2} + \log\frac{n}{\delta} + \log J}{2n-2}}$$

reverting from $\mathcal{G}$ back to $\mathcal{L}$, it means that:

$$\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\rho^2 A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)}[\mathcal{L}_D(\boldsymbol{\theta}+\boldsymbol{\epsilon})] \leq \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\rho^2 A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)}[\mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon})] + \sqrt{\frac{\frac{d}{2} + \frac{r^2}{2\rho^2} + \log\frac{n}{\rho} + \log J}{2n-2}}$$

Since $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \rho^2 A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)$, we can write $\boldsymbol{\epsilon} = A_{\boldsymbol{\theta}}\mathbf{z}$, in which $\mathbf{z} \sim \mathcal{N}(0, \rho^2 \mathbf{I}_d)$. Notice that $\|\boldsymbol{\epsilon}\|_2^2 = \boldsymbol{\epsilon}^\top\boldsymbol{\epsilon} = \mathbf{z}^\top A_{\boldsymbol{\theta}}^\top A_{\boldsymbol{\theta}}\mathbf{z} = \mathbf{z}^\top\mathbf{z} = \|\mathbf{z}\|_2^2$. We have the concentration inequality:

$$\mathbf{z} \sim \mathcal{N}(0, \rho^2\mathbf{I}_d) \implies \|\mathbf{z}\|^2 \leq d\rho^2\Big(1 + \sqrt{\frac{\log n}{d}}\Big)^2$$

with probability at least $1 - \frac{1}{\sqrt{n}}$. Thus, we have $\|\boldsymbol{\epsilon}\|^2 = \|\mathbf{z}\|_2^2 \leq d\rho^2(1 + \sqrt{\frac{\log n}{d}})^2$ with probability at least $1 - \frac{1}{\sqrt{n}}$. Denote $\gamma = \rho(\sqrt{d} + \sqrt{\log n})$, it follows that:

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\rho^2 A_{\boldsymbol{\theta}_j} A_{\boldsymbol{\theta}_j}^\top)}[\mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon})] &\leq \Big(1 - \frac{1}{\sqrt{n}}\Big) \max_{\|\boldsymbol{\epsilon}\|_2^2 \leq \gamma^2} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) + \frac{\mathcal{L}_{\max}}{\sqrt{n}} \\
&\leq \max_{\|\boldsymbol{\epsilon}\|_2^2 \leq \gamma^2} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) + \frac{\mathcal{L}_{\max}}{\sqrt{n}}
\end{aligned}$$

which implies

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\rho^2 A_{\boldsymbol{\theta}_j} A_{\boldsymbol{\theta}_j}^\top)}[\mathcal{L}_D(\boldsymbol{\theta}+\boldsymbol{\epsilon})] &\leq \max_{\|\boldsymbol{\epsilon}\|_2^2 \leq \gamma^2} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) + \frac{\mathcal{L}_{\max}}{\sqrt{n}} + \sqrt{\frac{\frac{d}{2} + \frac{r^2}{2\rho^2} + \log\frac{n}{\delta} + \log J}{2n-2}} \\
&\leq \max_{\|\boldsymbol{\epsilon}\|_2^2 \leq \gamma^2} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) + \frac{\mathcal{L}_{\max}}{\sqrt{n}} \\
&\quad + \sqrt{\frac{\frac{d}{2} + \frac{r^2(\sqrt{d}+\sqrt{\log n})^2}{2\gamma^2} + \log\frac{n}{\delta} + \log J}{2n-2}}
\end{aligned}$$

Now, we are left to bound $\log J$. Recall from our assumption that $R_j = \big\{\boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M} : (\boldsymbol{\epsilon} - \overline{\boldsymbol{\epsilon}}_j)^\top (A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)^* (\boldsymbol{\epsilon} - \overline{\boldsymbol{\epsilon}}_j) \leq r_j^2\big\}$, and since the tangent space also has d dimensions, so $\mathrm{vol}(R_j) = \mathcal{O}(r_j^d)$, which means $J = \mathcal{O}(\max_j \mathrm{diam}(\mathcal{M})^d / r_j^d)$, thus $\log J = \mathcal{O}(d)$. We derive that:

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\rho^2 A_{\boldsymbol{\theta}_j} A_{\boldsymbol{\theta}_j}^\top)}[\mathcal{L}_D(\boldsymbol{\theta}+\boldsymbol{\epsilon})] \\
&\leq \max_{\|\boldsymbol{\epsilon}\|_2^2 \leq \gamma^2, \boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) + \frac{\mathcal{L}_{\max}}{\sqrt{n}} + \sqrt{\frac{\frac{d}{2} + \frac{r^2(\sqrt{d}+\sqrt{\log n})^2}{2\gamma^2} + \log\frac{n}{\delta} + \log J}{2n-2}} \\
&\leq \max_{\|\boldsymbol{\epsilon}\|_2^2 \leq \gamma^2, \boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) + \sqrt{\frac{\frac{r^2(\sqrt{d}+\sqrt{\log n})^2}{2\gamma^2} + \log\frac{n}{\delta} + O(d)}{2n-2}}
\end{aligned}$$

Under the assumption that adding Gaussian perturbation on the weight space does not improve the test error, we have:

$$\begin{aligned}\mathcal{L}_D(\boldsymbol{\theta}) &\leq \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\rho^2 A_{\boldsymbol{\theta}} A_{\boldsymbol{\theta}}^\top)}[\mathcal{L}_D(\boldsymbol{\theta}+\boldsymbol{\epsilon})] \\ &\leq \max_{\|\boldsymbol{\epsilon}\|_2^2\leq\gamma^2, \boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) + \sqrt{\frac{\frac{r^2(\sqrt{d}+\sqrt{\log n})^2}{2\gamma^2} + \log\frac{n}{\delta} + \mathcal{O}(d)}{2n-2}}\end{aligned}$$

Since $\gamma \propto \rho$, by rescaling we can conclude that:

$$\mathcal{L}_D(\boldsymbol{\theta}) \leq \max_{\boldsymbol{\theta}'\in\mathcal{B}_{\boldsymbol{\theta}}^a(\rho;\mathcal{T})} \mathcal{L}_S(\boldsymbol{\theta}') + \sqrt{\frac{\frac{r^2(\sqrt{d}+\sqrt{\log n})^2}{2\rho^2} + \log\frac{n}{\delta} + \mathcal{O}(d)}{2(n-1)}}$$

□

A.1.5 Proof of Second inequality

(*The upper-bound on the manifold* $\mathcal{M}$): For any small $\rho > 0$ and $\delta \in [0;1]$, with a high probability $1-\delta$ over training set $\mathcal{S}$ generated from a distribution $\mathcal{D}$, the following holds:

$$\mathcal{L}_D(\boldsymbol{\theta}) \leq \max_{\boldsymbol{\theta}'\in\mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M})} \mathcal{L}_S(\boldsymbol{\theta}') + \mathcal{O}\left(C(\mathcal{M})\rho^2 + \sqrt{\frac{\frac{d}{\rho^2} + d + \log\frac{n}{\delta}}{n-1}}\right)$$

Proof. First, we recall the definition of the neighborhood on a manifold that $\mathcal{B}_{\boldsymbol{\theta}}(\rho,\mathcal{M}) = R_{\boldsymbol{\theta}}(\mathcal{B}_\rho^o(\boldsymbol{\theta},\mathcal{T}))$. Indeed, for any $\varepsilon > 0$, there exists $\boldsymbol{\epsilon}_1, \boldsymbol{\epsilon}_2 \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$ such that

$$\max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}, \|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) \geq \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_1) \geq \max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}, \|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) - \varepsilon$$

and also

$$\begin{aligned}\max_{\boldsymbol{\theta}'\in\mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M})} \mathcal{L}_S(\boldsymbol{\theta}') &= \max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}, \|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_S(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon})) \geq \mathcal{L}_S(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon})) \\ &\geq \max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}, \|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_S(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon}_2)) - \varepsilon.\end{aligned}$$

Combine the two inequalities together, we have the following:

$$\max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}, \|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) - \max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}, \|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_S(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon})) \leq \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_1) - \mathcal{L}_S(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon}_2)) + 2\varepsilon$$

in which $\boldsymbol{\epsilon}_1, \boldsymbol{\epsilon}_2 \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$ such that $\|\boldsymbol{\epsilon}_1\|_2^2, \|\boldsymbol{\epsilon}_2\|_2^2 \leq \rho^2$.

Under the assumption that the model space is a compact manifold, it means that the domain of $\nabla\mathcal{L}_{\mathcal{S}}$ is bounded. Therefore $\nabla\mathcal{L}_{\mathcal{S}}$ is bounded by a constant L, which follows that

$$\begin{aligned}\max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}, \|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}+\boldsymbol{\epsilon}) - \max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}, \|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_S(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon})) &\leq \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_1) - \mathcal{L}_S(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon}_2)) + 2\varepsilon \\ &\leq \Big(\mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_1) - \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_2)\Big) + \Big(\mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_2) - \mathcal{L}_S(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon}_2))\Big) + 2\varepsilon\end{aligned}$$

Regarding the first term of the inequality above, we have

$$\mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_1) - \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_2) = \nabla\mathcal{L}_S(C)(\boldsymbol{\epsilon}_1 - \boldsymbol{\epsilon}_2) \leq L\gamma$$

in which C in the segment connect two points, since $\theta + \varepsilon_1$ is a point in high dimensional space. According to Lemma 1 in Boumal et al. (2018), we can bound the second term as:

$$\mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}+\boldsymbol{\epsilon}_2) - \mathcal{L}_S(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon}_2)) \leq L\big\|R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon}_2) - \boldsymbol{\theta} - \boldsymbol{\epsilon}_2\big\|_F \leq LC(\mathcal{M})\|\boldsymbol{\epsilon}_2\|_2^2 \leq LC(\mathcal{M})\|\gamma\|_2^2.$$

It follows that

$$
\begin{aligned}
\mathcal{L}_{\mathcal{D}}(\boldsymbol{\theta}) &\leq \max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M},\|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}) + \sqrt{\frac{\mathcal{O}(\frac{d}{\rho^2}+d+\log\frac{n}{\delta})}{n-1}} \\
&\leq \max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M},\|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_{\mathcal{S}}(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon})) \\
&+ \Big[\max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M},\|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}+\boldsymbol{\epsilon}) - \max_{\boldsymbol{\epsilon}\in\mathcal{T}_{\boldsymbol{\theta}}\mathcal{M},\|\boldsymbol{\epsilon}\|_2^2\leq\rho^2} \mathcal{L}_{\mathcal{S}}(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon}))\Big] + \sqrt{\frac{\mathcal{O}(\frac{d}{\rho^2}+d+\log\frac{n}{\delta})}{n-1}} \\
&\leq \max_{\boldsymbol{\theta}'\in\mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M})} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}') + \sqrt{\frac{\mathcal{O}(\frac{d}{\rho^2}+d+\log\frac{n}{\delta})}{n-1}} + \varepsilon \\
&+ \Big[\mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_1) - \mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_2)\Big] + \Big[\mathcal{L}_S(\boldsymbol{\theta}+\boldsymbol{\epsilon}_2) - \mathcal{L}_S(R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon}_2))\Big] + \sqrt{\frac{\mathcal{O}(\frac{d}{\rho^2}+d+\log\frac{n}{\delta})}{n-1}} \\
&\leq \max_{\boldsymbol{\theta}'\in\mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M})} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}') + LC(\mathcal{M})\rho^2 + L\rho + \sqrt{\frac{\mathcal{O}(\frac{d}{\rho^2}+d+\log\frac{n}{\delta})}{n-1}} + 2\varepsilon \\
&\leq \max_{\boldsymbol{\theta}'\in\mathcal{B}_{\boldsymbol{\theta}}(\rho;\mathcal{M})} \mathcal{L}_{\mathcal{S}}(\boldsymbol{\theta}') + \mathcal{O}\left(C(\mathcal{M})\gamma^2 + \sqrt{\frac{\frac{d}{\rho^2}+d+\log\frac{n}{\delta}}{n-1}}\right)
\end{aligned}
$$

which concludes our proof. □

Lemma 1. *(Boumal et al., 2018) There exists a constant $C(\mathcal{M}) > 0$ such that for any $\boldsymbol{\theta} \in \mathcal{M}$ and $\boldsymbol{\epsilon} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}$, the following holds:*

$$\|R_{\boldsymbol{\theta}}(\boldsymbol{\epsilon}) - \boldsymbol{\theta} - \boldsymbol{\epsilon}\|_F \leq C(\mathcal{M})\,\|\boldsymbol{\epsilon}\|_F^2$$

The constant $C(\mathcal{M})$ value depends on the manifold structure and may scale with dimensions for general manifolds. However, for retractions on the Stiefel manifold, the constant is independent of (d, k) and can be computed explicitly. Specifically, when using the QR factorization or the polar decomposition as the retraction, we have $C(\mathcal{M}) = 1 + \sqrt{2}/2$.

A.2 Additional Experiments

A.2.1 Per-epoch runtime

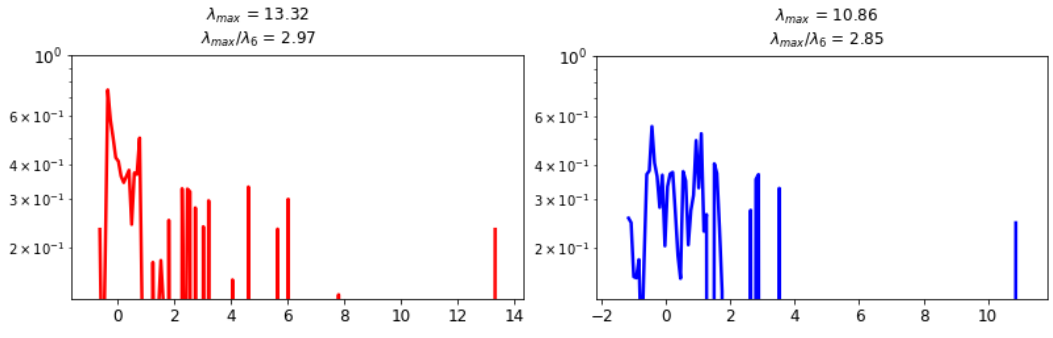
This ablation compares the single-epoch wallclock runtimes of SGD, SAM, and RSAM. Indeed, it is expected that SAM and RSAM take at least twice as long as SGD because both SAM and RSAM involve double backward-forward each iteration. We especially note that since RSAM involves additional computations on a manifold, it is expected that RSAM would take longer than SAM. As shown in Table 5, while RSAM improves the final performance, its runtime is only about 6% slower than SAM, therefore worth the tradeoff.

Method	**CIFAR100**		**CIFAR10**		**AirCraft**	
	ResNet34	ResNet50	ResNet34	ResNet50	ResNet34	ResNet50
SGD	$21.5_{\pm 1.73}$	$40.1_{\pm 2.96}$	$21.4_{\pm 1.72}$	$38.8_{\pm 3.05}$	$57.6_{\pm 2.59}$	$114.5_{\pm 4.77}$
SAM	$49.1_{\pm 1.68}$	$83.9_{\pm 2.79}$	$48.8_{\pm 1.62}$	$84.7_{\pm 2.68}$	$125.3_{\pm 1.3}$	$245.6_{\pm 4.30}$
RSAM	$52.6_{\pm 1.66}$	$88.7_{\pm 3.14}$	$51.1_{\pm 1.90}$	$88.3_{\pm 2.79}$	$133.0_{\pm 1.4}$	$259.2_{\pm 3.82}$

Table 5: Per-epoch wall-clock runtime in seconds.

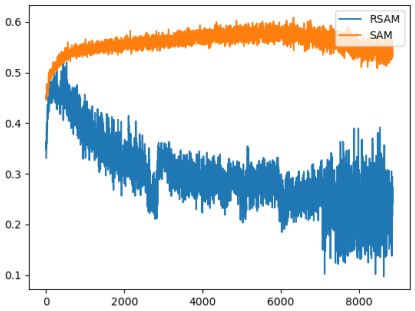
A.2.2 Sharpness Evolution and Hessian Spectral

Throughout this work, we have designed RSAM to actively seek local minima in the regions within a manifold with both low loss value and low sharpness. In this section, to further verify whether



(a) Hessian Spectra of SAM

(b) Hessian Spectra of RSAM



(c) Sharpness over iterations of SAM vs RSAM.

Figure 4: The spectrum of the Hessian at the termination of the training phase with SAM vs. RSAM (above) and the evolution of sharpness over iterations (below). The results are reported on the CIFAR100 dataset with the SupCon loss function. ρ in both methods is set to 0.1

RSAM found the low-sharpness region, we first contrast the spectral of the Hessian for ResNet34 trained on CIFAR100 for 400 steps with RSAM and SAM. Indeed, the model trained with RSAM has a lower maximum eigenvalue (10.86 of RSAM vs. 13.32 of SAM), and RSAM has a flatter eigenvalue distribution as shown in the Figure 4, therefore suggests that RSAM entered the lower-sharpness region on the loss landscape. Besides, we also report the sharpness evolutions over iterations as shown in Figure 4c. These results together indicate that RSAM successfully seeks points in lower-sharpness regions within the loss landscape.