Karcher Mean Weight Fusion: A Geometric Approach to Neural Network Model Merging

葉佐俊 (Zuojun-Ye)

2025 年 4 月 20 日

摘要

Model merging has emerged as a critical technique for combining the strengths of multiple neural networks without requiring retraining. This paper introduces a novel approach to weight fusion based on the Riemannian geometry concept of the Karcher mean. Unlike traditional linear interpolation methods, our approach treats neural network weights as points on a manifold and computes their intrinsic average. We present a robust algorithm that ensures numerical stability and demonstrate its theoretical properties. The proposed Karcher mean fusion method preserves the directional information in weight tensors while maintaining appropriate scaling, making it particularly suitable for merging models with different specializations. Our mathematical formulation provides a principled geometric framework for understanding and implementing model merging operations.

1 Introduction

The field of deep learning has seen remarkable progress in developing increasingly sophisticated neural network architectures. As these models grow in complexity and specialization, there is growing interest in techniques that can combine multiple models to create more generalized and robust systems. Model merging offers an efficient alternative to ensemble methods by fusing multiple models into a single model that inherits the strengths of its constituents without incurring additional inference costs.

Traditional model merging approaches often rely on simple weighted averaging of parameters [2, 3], which treats the weight space as Euclidean. However, there is increasing evidence that the loss landscape of neural networks exhibits a non-Euclidean geometry [4]. This suggests that more sophisticated geometric approaches might yield better results when combining models.

In this paper, we introduce a novel approach to model merging based on the Karcher mean [1], a generalization of the arithmetic mean to Riemannian manifolds. Our approach treats weight tensors as points on a manifold and computes their intrinsic average, taking into account the curved geometry of the space. This geometric perspective leads to several desirable properties:

- Preservation of directional information in weight tensors
- Appropriate scaling based on the magnitude of constituent weights
- Numerical stability even for orthogonal or nearly orthogonal weight vectors
- A principled framework that generalizes traditional linear interpolation

We provide a detailed mathematical formulation of our approach, along with an efficient algorithm for computing the Karcher mean of multiple weight tensors. Our method is applicable to a wide range of neural network architectures and can handle different model formats, including both continuous and discrete parameters.

2 Related Work

2.1 Model Averaging

Model averaging has been studied extensively in machine learning. Izmailov et al. [2] proposed Stochastic Weight Averaging (SWA), which averages the weights of models obtained during training to improve generalization. Wortsman et al. [3] extended this idea to model soups, which blend multiple fine-tuned models to obtain improved performance.

2.2 Geometric Methods in Deep Learning

The geometric perspective on deep learning has gained traction in recent years. Li et al. [4] developed techniques for visualizing the loss landscape of neural networks, revealing its complex geometric structure. Amari et al. [5] proposed natural gradient methods, which leverage the information geometry of the parameter space to improve optimization.

2.3 Riemannian Geometry in Machine Learning

Riemannian geometry has found numerous applications in machine learning. Nickel and Kiela [6] used hyperbolic embeddings for hierarchical data. Bonnabel [7] developed stochastic gradient descent on Riemannian manifolds. However, the application of Riemannian geometry to model merging remains underexplored.

3 Mathematical Background

3.1 Riemannian Manifolds

A Riemannian manifold is a smooth manifold M equipped with a Riemannian metric, which assigns an inner product to the tangent space at each point. This allows for the notion of distance, angles, and volume on the manifold.

3.2 Geodesics

Geodesics are the generalization of straight lines to curved spaces. On a Riemannian manifold, they represent the shortest paths between points. Mathematically, geodesics satisfy the geodesic equation:

$$\frac{d^2x^i}{dt^2} + \Gamma^i_{jk}\frac{dx^j}{dt}\frac{dx^k}{dt} = 0 \tag{1}$$

where Γ^i_{jk} are the Christoffel symbols that encode the curvature of the space.

3.3 Karcher Mean

The Karcher mean (also known as the Fréchet mean or geometric mean) of a set of points $\{x_1, x_2, \ldots, x_N\}$ on a Riemannian manifold is defined as:

$$\mu = \arg\min_{x \in M} \sum_{i=1}^{N} \alpha_i d^2(x, x_i) \tag{2}$$

where $d(x, y)$ is the geodesic distance between points x and y, and α_i are non-negative weights that sum to 1. The Karcher mean generalizes the notion of the weighted arithmetic mean to curved spaces.

4 Karcher Mean Weight Fusion

4.1 Problem Formulation

Given a set of neural network models $\{M_1, M_2, \ldots, M_N\}$ with corresponding weight tensors $\{W_1, W_2, \ldots, W_N\}$ and weights $\{\alpha_1, \alpha_2, \ldots, \alpha_N\}$ such that $\sum_{i=1}^{N} \alpha_i = 1$, we aim to compute a merged model M with weight tensors W that effectively combines the knowledge and capabilities of the individual models.

We treat each weight tensor as a point on a manifold and compute their Karcher mean, taking into account both the direction and magnitude of the tensors.

4.2 Directional and Magnitude Components

We decompose each weight tensor W_i into its magnitude $\|W_i\|$ and direction $U_i = \frac{W_i}{\|W_i\|}$:

$$W_i = \|W_i\| \cdot U_i \tag{3}$$

The directional component U_i lies on the unit sphere, which is a well-studied Riemannian manifold. The magnitude component $\|W_i\|$ is a scalar that represents the overall scale of the weights.

4.3 Karcher Mean on the Unit Sphere

To compute the Karcher mean of the directions $\{U_1, U_2, \ldots, U_N\}$, we use an iterative algorithm based on the exponential and logarithmic maps on the sphere.

The logarithmic map at point p takes a point q on the sphere and returns a tangent vector v such that a geodesic starting at p in the direction of v reaches q after unit time:

$$\log_p(q) = \frac{\arccos(p \cdot q)}{\sin(\arccos(p \cdot q))}(q - (p \cdot q)p) \tag{4}$$

The exponential map at point p takes a tangent vector v and returns the point q reached by following the geodesic starting at p in the direction of v for unit time:

$$\exp_p(v) = \cos(\|v\|)p + \sin(\|v\|)\frac{v}{\|v\|} \tag{5}$$

Using these maps, we can compute the Karcher mean through the following iterative algorithm:

4.4 Weighted Magnitude

For the magnitude component, we use a weighted arithmetic mean:

$$\|W\| = \sum_{i=1}^{N} \alpha_i \|W_i\| \tag{6}$$

4.5 Merged Weights

The final merged weight tensor is obtained by combining the directional and magnitude components:

$$W = \|W\| \cdot U \tag{7}$$

This approach ensures that the merged weights preserve both the directional information and the overall scale of the original weights.

Algorithm 1 Karcher Mean on the Unit Sphere

Input: Unit vectors $\{U_1, U_2, \ldots, U_N\}$, weights $\{\alpha_1, \alpha_2, \ldots, \alpha_N\}$, maximum iterations max_iter, tolerance ϵ
Output: Karcher mean U
Initialize U as the normalized weighted arithmetic mean of $\{U_1, U_2, \ldots, U_N\}$
for $iter = 1$ to max_iter **do**
    $T \leftarrow \mathbf{0}$
    for $i = 1$ to N **do**
        $dot \leftarrow \text{clamp}(U \cdot U_i, -1, 1)$
        $\theta \leftarrow \arccos(dot)$
        if $\theta < \epsilon$ **then**
            continue
        end if
        $T \leftarrow T + \alpha_i \frac{\theta}{\sin(\theta)}(U_i - dot \cdot U)$
    end for
    $norm\_T \leftarrow \|T\|$
    if $norm\_T < \epsilon$ **then**
        break
    end if
    $U \leftarrow \cos(norm\_T) \cdot U + \sin(norm\_T) \cdot \frac{T}{norm\_T}$
    Normalize U to ensure $\|U\| = 1$
end for
return U

5 Theoretical Analysis

5.1 Convergence Properties

The iterative algorithm for computing the Karcher mean on the unit sphere is guaranteed to converge under mild conditions. Specifically, if the input points are contained within a geodesic ball of radius less than $\frac{\pi}{2}$, the Karcher mean exists and is unique, and our algorithm will converge to it [8].

5.2 Relationship to Linear Interpolation

When the weight tensors are nearly collinear (i.e., pointing in similar directions), our Karcher mean approach approximates linear interpolation. This can be seen by considering the small angle approximation of the geodesic distance on the sphere.

5.3 Handling of Orthogonal Weights

One of the key advantages of our approach is its handling of orthogonal or nearly orthogonal weight vectors. Traditional linear interpolation can result in a merged vector with small magnitude when the input vectors are orthogonal. In contrast, our Karcher mean approach finds a direction that minimizes the geodesic distance to all input directions, ensuring that the merged weights maintain a meaningful direction.

6 Implementation Details

6.1 Numerical Stability

To ensure numerical stability, we incorporate several techniques:

- Clamping the dot product to the range $[-1, 1]$ to avoid numerical issues when computing the arc cosine
- Special handling of small angles to avoid division by zero
- Ensuring that the result is always a unit vector by explicit normalization

6.2 Handling Tensors of Different Shapes

Neural network weights often have different shapes, particularly when merging models with different architectures. We address this by first resizing the tensors to a common shape using a zero-padding approach.

6.3 Support for Different File Formats

Our implementation supports both .safetensors and .bin file formats, allowing for seamless integration with popular deep learning frameworks.

7 Mathematical Formulation

Let's formalize our Karcher mean weight fusion approach in mathematical terms.

Given weight tensors $W_1, W_2, \ldots, W_N \in \mathbb{R}^d$ and corresponding weights $\alpha_1, \alpha_2, \ldots, \alpha_N$ such that $\sum_{i=1}^{N} \alpha_i = 1$, we decompose each tensor into its magnitude and direction:

$$W_i = \|W_i\| \cdot U_i \tag{8}$$

where $\|W_i\| = \sqrt{\sum_{j=1}^{d} (W_i)_j^2}$ and $U_i = \frac{W_i}{\|W_i\|}$.

The Karcher mean U of the directions $U_1, U_2, \ldots, U_N$ is the solution to the minimization problem:

$$U = \arg\min_{V:\|V\|=1} \sum_{i=1}^{N} \alpha_i d^2(V, U_i) \tag{9}$$

where $d(V, U_i) = \arccos(V \cdot U_i)$ is the geodesic distance on the unit sphere.

This minimization problem can be solved using gradient descent on the manifold. The gradient of the squared geodesic distance with respect to V is given by:

$$\nabla_V d^2(V, U_i) = -2 \frac{\arccos(V \cdot U_i)}{\sin(\arccos(V \cdot U_i))} (U_i - (V \cdot U_i)V) \tag{10}$$

Taking a step in the negative gradient direction and projecting back onto the unit sphere leads to the update rule:

$$V_{t+1} = \frac{\exp_{V_t}\left(-\eta \sum_{i=1}^{N} \alpha_i \nabla_{V_t} d^2(V_t, U_i)\right)}{\| \exp_{V_t}\left(-\eta \sum_{i=1}^{N} \alpha_i \nabla_{V_t} d^2(V_t, U_i)\right) \|} \tag{11}$$

where η is the step size and exp is the exponential map on the sphere.

After computing the Karcher mean direction U, we combine it with the weighted magnitude:

$$W = \left(\sum_{i=1}^{N} \alpha_i \|W_i\|\right) \cdot U \tag{12}$$

This gives us the final merged weight tensor that incorporates both the directional and magnitude information from the original weight tensors.

8 Discussion

8.1 Advantages Over Traditional Methods

The Karcher mean weight fusion approach offers several advantages over traditional model merging methods:

- **Geometric Interpretation:** By treating weight tensors as points on a manifold, our approach aligns with the geometric nature of neural network optimization.
- **Directional Preservation:** The Karcher mean preserves the directional information in the weight tensors, ensuring that the merged model maintains the functional properties of the original models.
- **Handling of Orthogonal Weights:** Our approach can effectively merge weights that are orthogonal or nearly orthogonal, a case where linear interpolation might produce suboptimal results.
- **Theoretical Foundation:** The Karcher mean has a solid mathematical foundation in Riemannian geometry, providing a principled framework for model merging.

8.2 Potential Applications

The Karcher mean weight fusion approach has potential applications in various areas:

- **Knowledge Distillation:** Merging multiple teacher models to create a more effective teacher for knowledge distillation.
- **Multi-Task Learning:** Combining models trained on different tasks to create a model that can perform multiple tasks.
- **Federated Learning:** Aggregating models trained on different devices or data sources in a privacy-preserving manner.
- **Model Adaptation:** Adapting pre-trained models to specific domains or tasks without extensive retraining.

8.3 Limitations and Future Work

While our approach offers significant advantages, it also has some limitations:

- **Computational Complexity:** Computing the Karcher mean can be more computationally intensive than simple linear interpolation, especially for large models.
- **Applicability to Non-Euclidean Parameters:** Some model parameters, such as discrete embeddings, may not be naturally represented as points on a Euclidean or spherical manifold.
- **Handling of Complex Architectures:** Merging models with significantly different architectures remains a challenge.

Future work could explore extensions to handle these limitations, as well as empirical evaluations on a wider range of tasks and model architectures.

9 Conclusion

In this paper, we have introduced a novel approach to model merging based on the Karcher mean, a geometric concept from Riemannian geometry.

Our approach treats weight tensors as points on a manifold and computes their intrinsic average, taking into account both the direction and magnitude of the weights. We have provided a detailed mathematical formulation, an efficient algorithm, and a theoretical analysis of our approach.

The Karcher mean weight fusion approach offers a principled geometric framework for model merging, with advantages over traditional linear interpolation methods, particularly in handling orthogonal or nearly orthogonal weights. We believe this approach opens up new possibilities for combining neural networks in a way that respects the geometric structure of their parameter spaces.

参考文献

[1] Karcher, H. (1977). Riemannian center of mass and mollifier smoothing. *Communications on Pure and Applied Mathematics*, 30(5):509–541.

[2] Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D., and Wilson, A. G. (2018). Averaging weights leads to wider optima and better generalization. In *Proceedings of the 34th Conference on Uncertainty in Artificial Intelligence (UAI)*.

[3] Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., et al. (2022). Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning (ICML)*.

[4] Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. (2018). Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems (NeurIPS)*.

[5] Amari, S.-I. (1998). Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276.

[6] Nickel, M. and Kiela, D. (2017). Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems (NeurIPS)*.

[7] Bonnabel, S. (2013). Stochastic gradient descent on Riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229.

[8] Afsari, B. (2011). Riemannian L^p center of mass: Existence, uniqueness, and convexity. *Proceedings of the American Mathematical Society*, 139(2):655–673.