Multi-Model Merging via Spherical Barycenters (SLERP for $\mathcal{N} > 2$)

January 15, 2026

Francesca Maggiore

1. INTRODUCTION

Il Model Merging negli ultimi anni è diventata un'ottima strategia per combinare modelli pre-addestrati piuttosto che crearne sempre di nuovi, riducendo notevolmente i costi di calcolo. In questo progetto è stato definito un metodo di merging geometrico valido per $\mathcal{N} > 2$ modelli.

1.1. Model Merging

Il model merging riesce a combinare più modelli addestrati, checkpoint, fra di loro in un unico modello finale, in grado di apprendere le abilità individuali di ognuno senza la necessità di un fine-tuning o training aggiuntivo. Questo è molto utile perché addestrare un modello da zero per ogni abilità risulta dispendioso in termini di memoria e tempo. Molto spesso, inoltre, il training da zero non garantisce un miglioramento; con il merging, invece, si sfruttano i pesi già ottimizzati, diminuendo i costi e l'incertezza del risultato.

1.2. Limiti delle Media Euclidea

Per riuscire in questa "fusione", la semplice media aritmetica non basta perché assume erroneamente che lo spazio dei pesi sia Euclideo. Questo approccio porta spesso a diminuire l'accuratezza dei modelli fusi. Bisogna invece considerare che i pesi normalizzati delle reti neurali vivono su una superficie curva, detta Manifold sferica. Calcolare la media classica in questo spazio significa "tagliare" attraverso la sfera, finendo in zone dove il modello non funziona bene.

1.3. Geometria sulle Manifold e Karcher mean

Per rispettare questa geometria curva, viene utilizzato il Karcher Mean. A differenza della media lineare, il Karcher Mean calcola il punto centrale muovendosi lungo la superficie della sfera, seguendo le linee curve o geodetiche. Questo metodo è l'estensione dello SLERP ed è l'approccio matematicamente corretto per fondere $\mathcal{N}$ modelli preservandone le caratteristiche geometriche.

$$\mathbf{m}^* = \arg\min_{\mathbf{m} \in \mathcal{S}^{D-1}} \sum_{i=1}^{N} w_i \cdot d^2_{geo}(\mathbf{m}, \mathbf{v}_i) \tag{1}$$

1.4. Permutation Invariance

Prima di applicare il Karcher Mean, bisogna risolvere un ultimo problema: la Permutation Invariance. Nelle reti neurali, l'ordine dei neuroni è spesso diverso da un modello all'altro. Se si prova a fare la media di modelli non allineati, si crea una forte interferenza distruttiva che fa crollare l'accuratezza per $\mathcal{N}$ grande. Per evitare questo, bisogna integrare una fase di Allineamento che riordina i neuroni in base alla loro similarità prima di calcolare il punto medio. (Ainsworth et al., 2023)

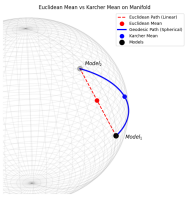

Figure 1. La Media Euclidea (rosso) "taglia" la manifold uscendo dallo spazio dei pesi validi, mentre il Karcher Mean (blu) segue la geodetica sulla superficie sferica $\mathcal{S}^{D-1}$.

2. RELATED WORK

In questa sezione si analizza l'evoluzione del problema di combinare più reti neurali, partendo dai metodi classici per arrivare alle tecniche geometriche.

2.1. Deep Ensemble

Solitamente per migliorare le prestazioni si usano i Deep Ensembles: si prendono le predizioni di tanti modelli diversi e si calcola la media. Funziona bene, ma richiede di salvare ed eseguire N modelli contemporaneamente, occupando troppa memoria e potenza di calcolo. Il Model Merging nasce proprio per risolvere questo problema. Tut-

Email: Francesca Maggiore <maggiore.2154286@studenti.uniroma1.it>.

Machine Learning 2025, Sapienza University of Rome, 2nd semester a.y. 2024/2025.

tavia, come precedentemente visto, la fusione tramite media Euclidea spesso fallisce a causa della complessità della funzione di Loss. (Lakshminarayanan et al., 2017)

2.2. SLERP

Per risolvere i problemi della media lineare, si usa spesso la Spherical Linear Interpolation (SLERP). Questa tecnica permette di interpolare correttamente due modelli muovendosi sulla superficie sferica dei pesi invece che tagliare attraverso di essa. Il limite principale di SLERP, però, è che funziona solo per coppie di modelli. L'utilizzo del Karcher Mean serve proprio a estendere questa intuizione geometrica a un gruppo di modelli arbitrario. (Shoemake, 1985)

3. METHOD

In questo paragrafo viene descritta la pipeline sviluppata per il merging geometrico di $\mathcal{N}$ reti neurali. Il processo segue tre fasi sequenziali: proiezione sulla manifold, allineamento strutturale e ottimizzazione sferica.

3.1. Vettorizzazione e Proiezione

Il primo passo consiste nel trasformare i checkpoint PyTorch in una rappresentazione matematica trattabile. Vengono concatenati tutti i parametri, pesi e bias, di ogni modello in un unico vettore unidimensionale. Successivamente, applichiamo una normalizzazione $\mathcal{L}_2$ a ciascun vettore. Questo passaggio è fondamentale perché proietta i modelli, che inizialmente risiedono in uno spazio piatto, sulla superficie di una Manifold Sferica. Da questo momento in poi, i pesi non sono più semplici numeri, ma diventano punti su un'ipersfera.

3.2. Allineamento dei Pesi

Prima di procedere alla fusione, è necessario risolvere la Permutation Invariance. Poiché le reti neurali possono avere neuroni in ordine diverso pur svolgendo la stessa funzione, la media diretta risulterebbe incoerente. Si adeoota quindi una strategia di allineamento: fissiamo il primo modello come riferimento e riordiniamo i neuroni degli strati nascosti, degli altri modelli, affinché corrispondano il più possibile a quelli del riferimento. Questo massimizza la similarità tra i vettori prima del calcolo della media.

3.3. Calcolo del Karcher Mean

Una volta allineati i modelli sulla sfera, si calcola il loro baricentro geometrico. Poiché non esiste una formula diretta per $\mathcal{N} > 2$, viene utilizzato un algoritmo iterativo di discesa del gradiente. Ad ogni passo, il punto medio viene spostato verso la soluzione ottimale e si applica immediatamente una operazione di rinormalizzazione. Questo garantisce che il modello finale rimanga vincolato alla geometria sferica, preservando le proprietà statistiche dei pesi originali.

4. RESULTS

Per validare il metodo, si addestrano modelli indipendenti di tipo SimpleMLP sul dataset MNIST e si analizza il comportamento dell'algoritmo al variare di $\mathcal{N}$.

4.1. Analisi dell'Allineamento

Sono stati condotti i test in modo incrementale. In una prima fase, con $\mathcal{N} = 3$, l'algoritmo ha mostrato subito buone prestazioni. Tuttavia, aumentando l'esperimento a $\mathcal{N} = 10$, è stato riscontrato un drastico calo dell'accuratezza intorno al 10%. Il problema risiedeva nella forte interferenza distruttiva generata, dal disallineamento dei pesi. Infatti dopo aver integrato la fase di Allineamento i risultati sono migliorati notevolmente, riportando l'accuratezza sopra il 95%. Questo conferma che gestire le simmetrie di permutazione è indispensabile quando si fondono tanti modelli.

4.2. Risultati finali

La tabella seguente mostra i risultati finali. Si nota chiaramente che il Karcher Mean superi sempre la semplice media lineare. Si osserva anche che, all'aumentare di $\mathcal{N}$, l'approccio geometrico si dimostra più robusto, marcando una differenza con i risultati dati dall'approccio Euclideo.

Table 1. Confronto delle prestazioni su MNIST al variare di N.

N Models	Karcher Mean	Euclidean Mean	Gap
3	96.72%	96.65%	+0.07%
5	96.32%	96.14%	+0.18%
10	95.64%	95.43%	+0.21%

5. CONCLUSIONS

I risultati confermano che modellare lo spazio dei pesi come una Manifold Sferica è superiore all'approccio Euclideo. Teoricamente, per $\mathcal{N} \to \infty$, la media lineare tenderebbe a "collassare" verso l'origine, mentre il Karcher Mean preserva la norma e l'energia del segnale: il valore crescente del Gap in Tabella 1 conferma il limite dell'approccio lineare. Tuttavia, la geometria da sola non basta: l'Allineamento dei neuroni è un passaggio fondamentale. Senza risolvere la Permutation Invariance, l'interferenza distruttiva annulla i benefici geometrici, rendendo impossibile il merging su larga scala.

References

Ainsworth, S., Hayase, J., and Srinivasa, S. Git re-basin: Merging models modulo permutation symmetries. In *International Conference on Learning Representations (ICLR)*, 2023.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Rodolà, E. Dispense del corso: Multi-layer perceptrons, riduzione dimensionale e metodi ensemble, 2024. Materiale Didattico Universitario.

Shoemake, K. Animating rotation with quaternion curves. *SIGGRAPH Computer Graphics*, 19(3):245–254, 1985.