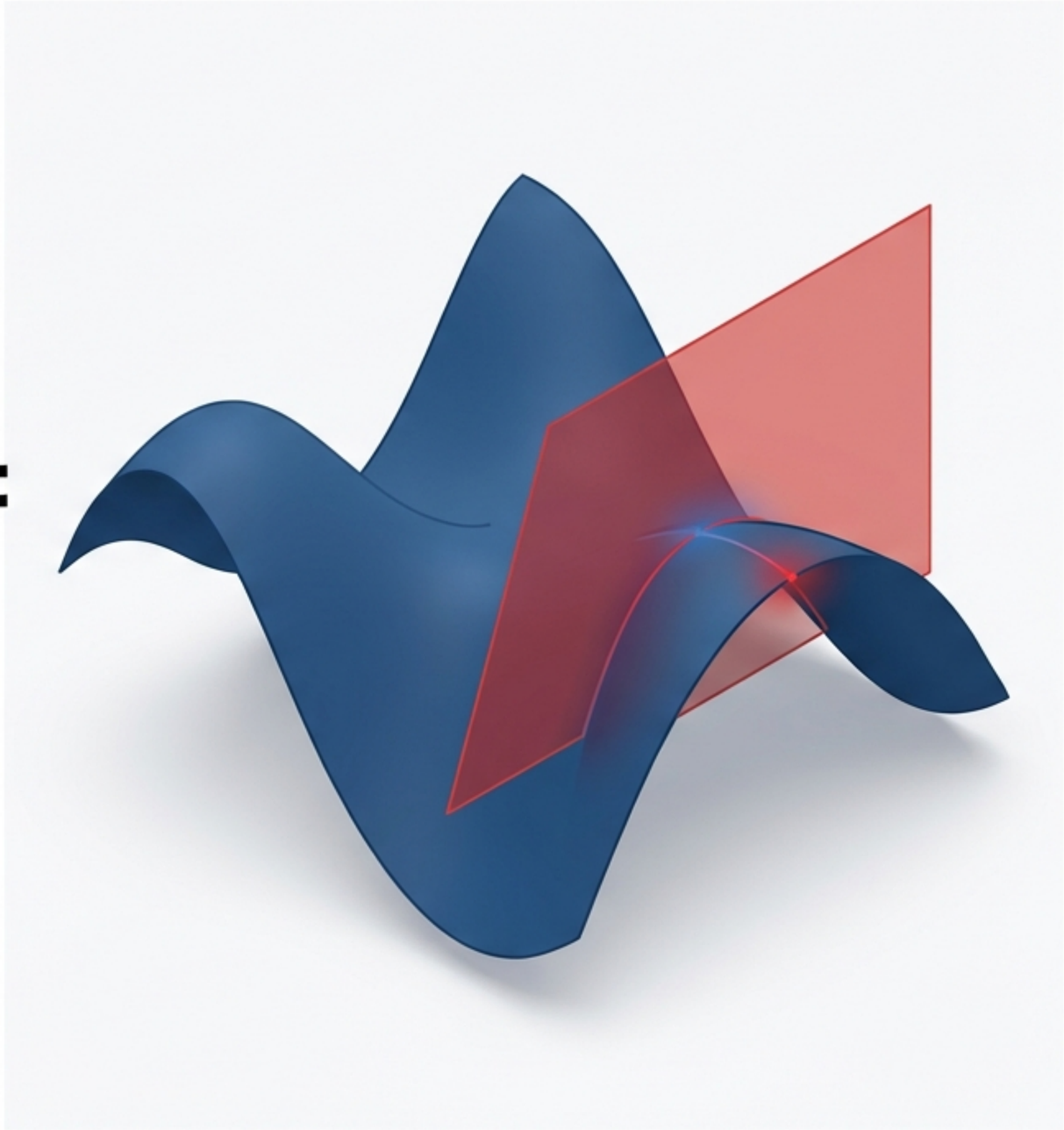
THINKING MACHINES LAB / JEREMY BERNSTEIN

Modular Manifolds: การปฏิวัติการหาค่าเหมาะที่สุดด้วยเรขาคณิต

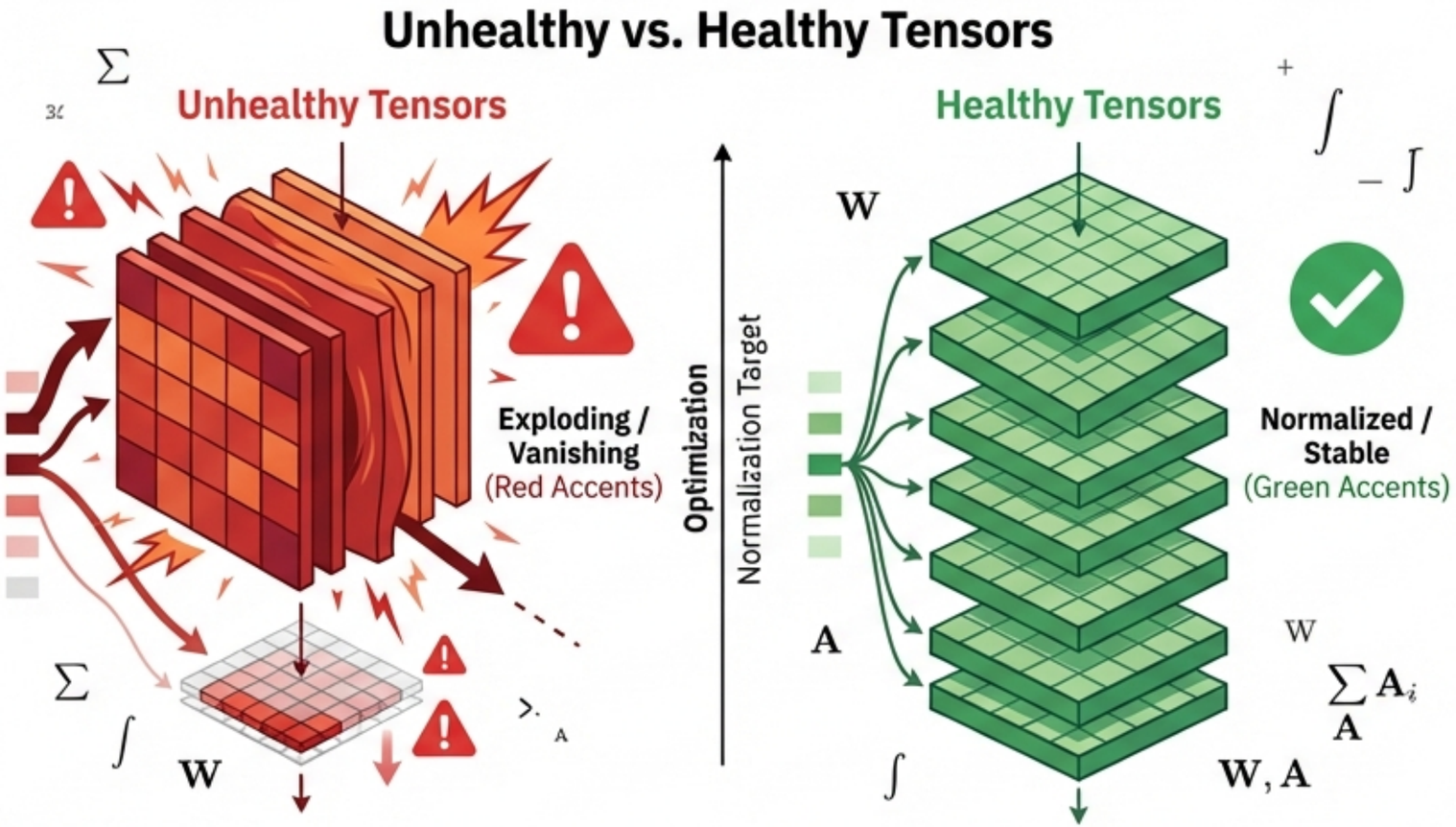
แนวทางใหม่ในการควบคุม Weight Matrices
เพื่อเสถียรภาพของโครงข่ายประสาทเทียม



Modular Manifolds: Rethinking Optimization with Geometry

NotebookLM

ปัญหาความไม่สมดุลของ Tensor ในการเทรนขนาดใหญ่



เมื่อเทรนโครงข่ายขนาดใหญ่ เราต้องการให้ Tensors **'มีสุขภาพดี'** (Healthy) คือไม่ใหญ่หรือเล็กจนเกินไป

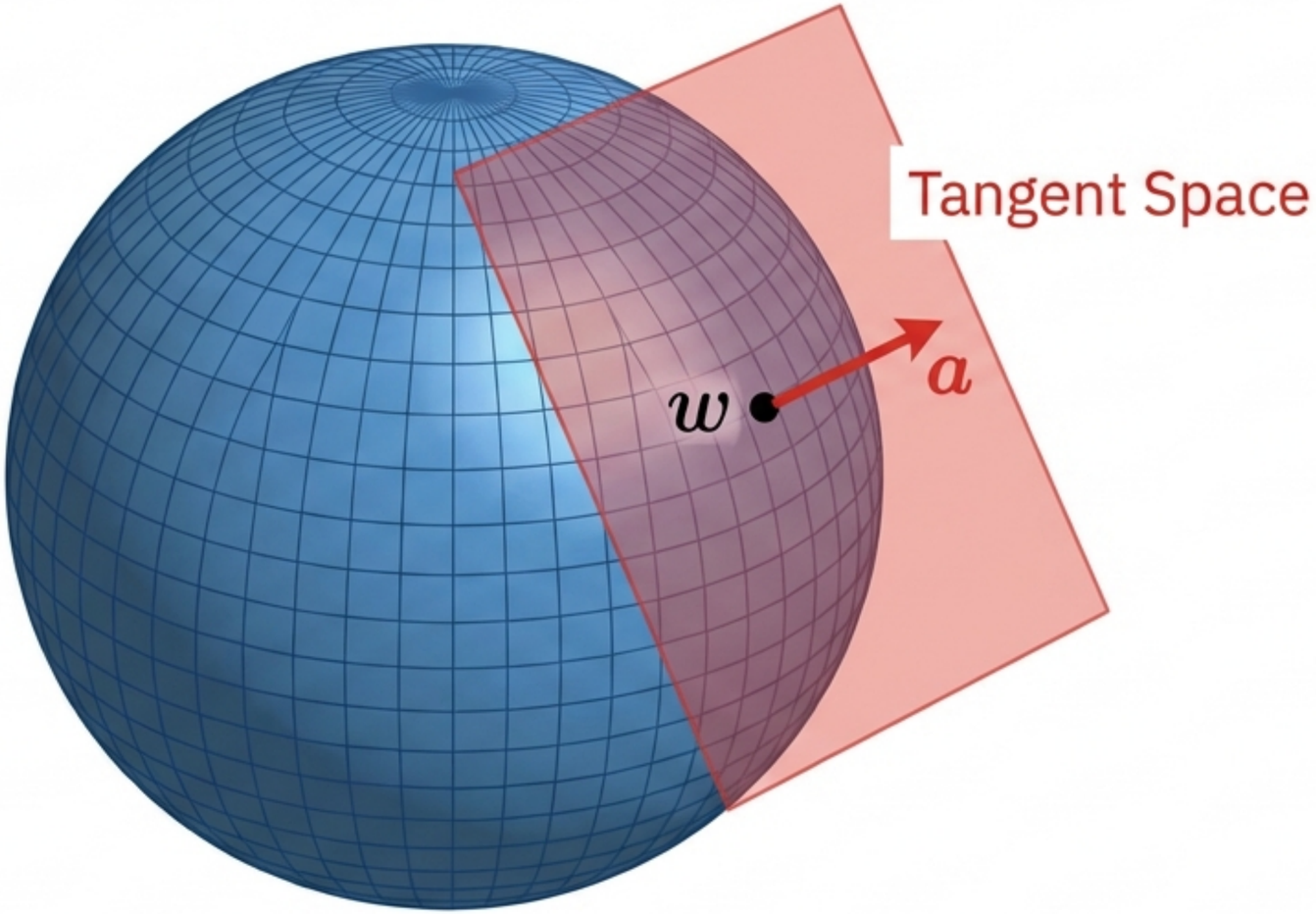
Current Standard: เราใช้ Normalization กับ Activations (เช่น LayerNorm) และ Gradients แต่สิ่งหนึ่งที่ขาดหายไปคือ...

The Missing Piece: การทำ Normalization กับ **Weight Matrices** โดยตรง

การกำหนดข้อจำกัด (Constraints) ให้กับ Weights จะช่วยควบคุม Condition Number ให้พฤติกรรมของโมเดลคาดเดาได้

NotebookLM

ภาพจำลองของแมนิโฟลด์: พื้นผิวโค้งและพื้นที่สัมผัส



Manifold
พื้นผิวโค้งที่เมื่อซูมเข้าไปใกล้ๆ จะดูเหมือนพื้นราบ (Locally flat)

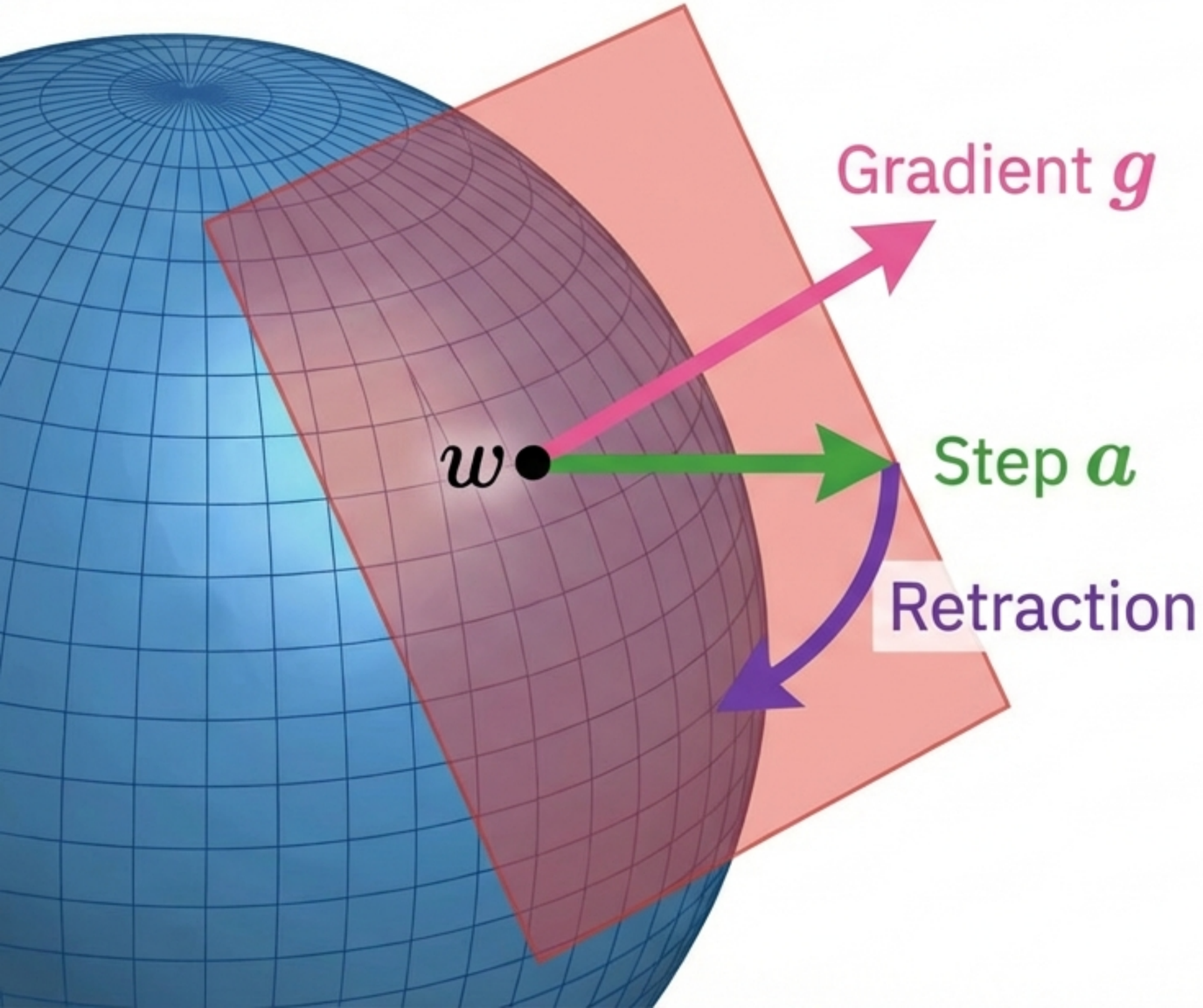
Tangent Space
พื้นที่ราบเสมือน ณ จุดจุดหนึ่งบนแมนิโฟลด์

$$w \in \mathbb{R}^d, \|\boldsymbol{w}\|_2 = 1$$

$$\boldsymbol{a} \perp \boldsymbol{w}$$

NotebookLM

กลไกการเคลื่อนที่: Step และ Retraction



1. **Tangent Step:**
   หาเจาะจงทิศทางที่ดีที่สุดใน Tangent Space (พื้นที่ราบ)

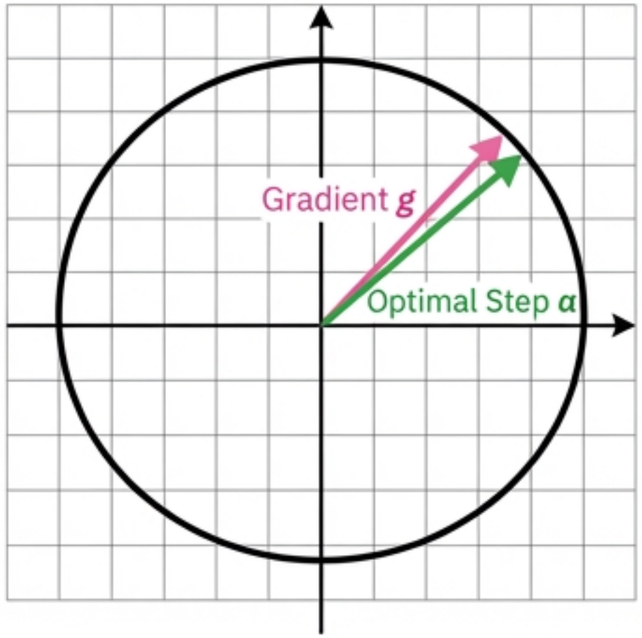
2. **Update:**
   ขยับไปตามทิศทางนั้นเป็นระยะทาง η (Learning Rate)

3. **Retract:**
   ดึงค่าที่ได้กลับลงมาบนพื้นผิวแมนิโฟลด์ (Project back)

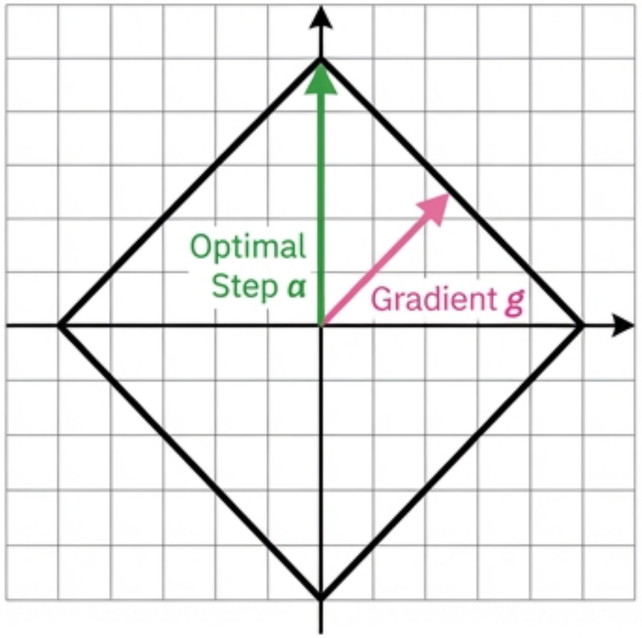
NotebookLM

เรขาคณิตเป็นตัวกำหนดทิศทางที่ดีที่สุด

Euclidean Norm



Non-Euclidean Norm



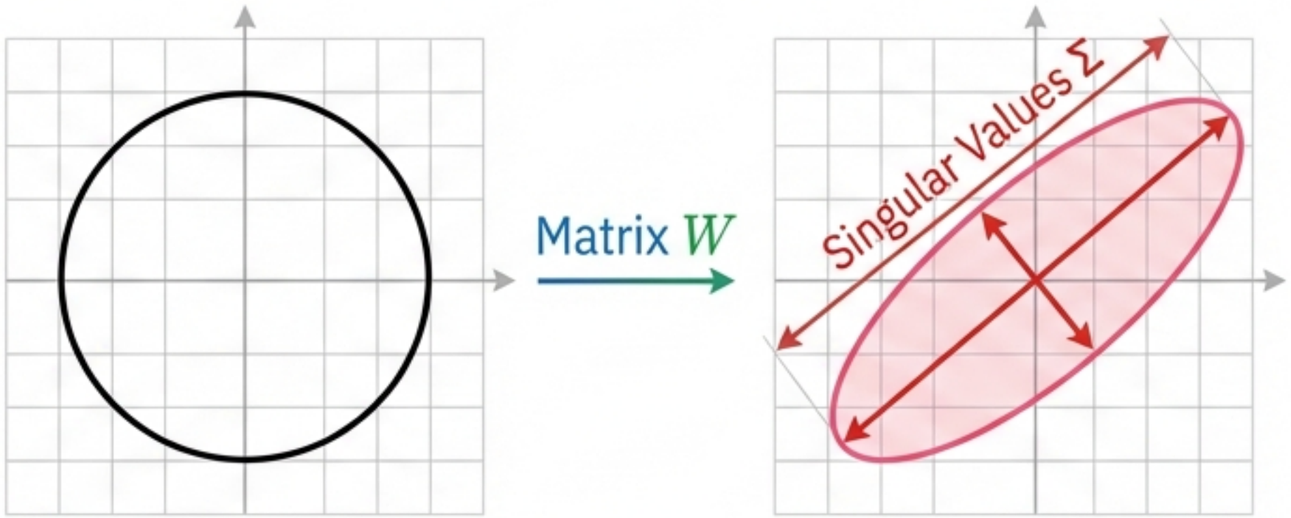
- ทิศทางของ Gradient (ลูกศรชมพู) อาจไม่ใช่ทิศทางที่เราควรเดินเสมอไป

รูปร่างของ ‘Unit Ball’ เป็นตัวกำหนดว่า เราจะเดินไปทางไหนได้ไกลที่สุดเพื่อลด Loss

$$\min_{\mathbf{a}} \mathbf{a}^\top \mathbf{g} \text{ subject to } \|\mathbf{a}\| = \eta$$

NotebookLM

จากเวกเตอร์สู่เมทริกซ์: การควบคุมการยืดขยาย



ใน Transformer, Weights คือ Matrix (W) ที่แปลง Input Vector (x) เป็น Output ($y = Wx$)

เราต้องการให้ Matrix มีพฤติกรรมที่ ‘เป็นกลาง’ ไม่ยืด (Stretch) หรือหดข้อมูลมากเกินไป

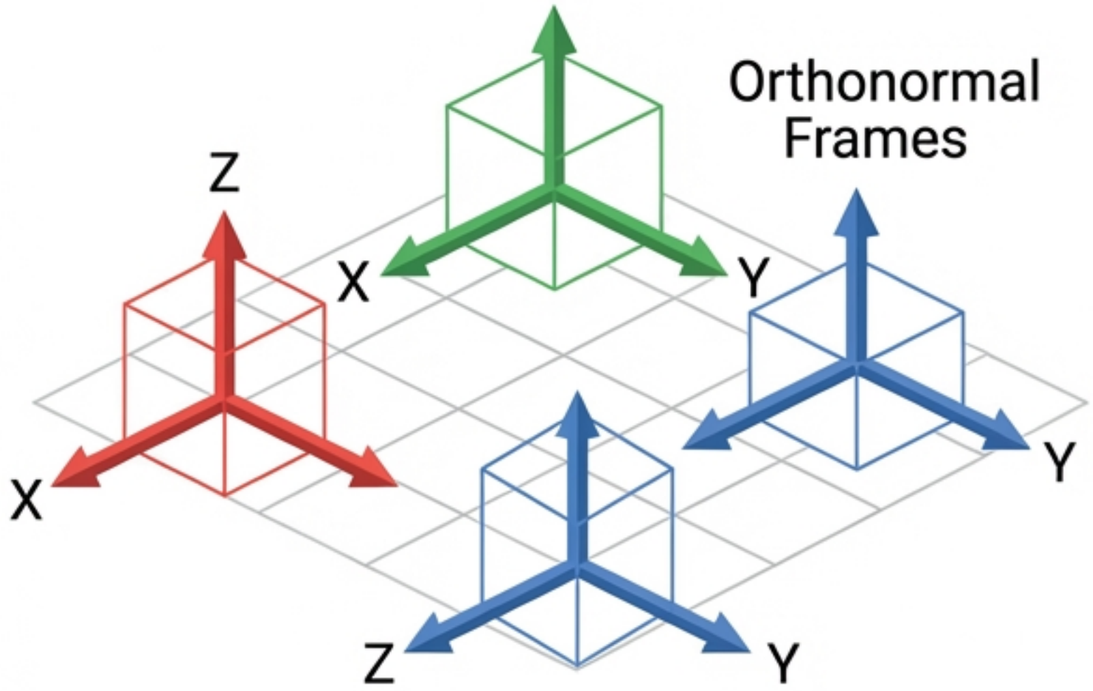
เครื่องมือวัดคือ **Singular Value Decomposition (SVD)**

NotebookLM

รู้จักกับ Stiefel Manifold

Stiefel Manifold: แมนิโฟลด์ของ Matrix ที่มี Singular Values ทั้งหมดเท่ากับ 1

$$\mathrm{Stiefel}(m,n) := \{W \in \mathbb{R}^{m\times n} \mid W^T W = I_n\}$$



Tangent Space Condition

$$A^\top W + W^\top A = 0$$

เงื่อนไขนี้คือการขยายแนวคิดจากทรงกลมมาสู่โลกของเมทริกซ์

NotebookLM

Manifold Muon: อัลกอริทึมสำหรับเมทริกซ์

การผสมผสานระหว่างข้อจำกัดทางเรขาคณิต (Stiefel) และบรรทัดฐานการวัดระยะ (Spectral Norm)

$$\min_{A} \operatorname{trace}(G^T A)$$ ← Minimize Loss Change

Subject to:

$$\|A\|_{\text{spectral}} \leq \eta$$ ← Spectral Size Constraint (Muon)

$$A^T W + W^T A = 0$$ ← Tangent Constraint (Stiefel)

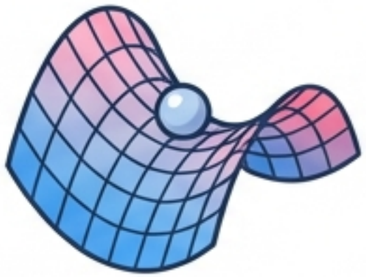
NotebookLM

การแก้ปัญหาด้วย Dual Ascent

Step 1: Primal (Hard)

Constrained Minimization

Step 2: Saddle Point



Min-Max Lagrange

Step 3: Dual (Solvable)

Unconstrained Maximization
(Gradient Ascent)

$$\max_{\boldsymbol{\Lambda}} - \eta \times \|\boldsymbol{G} + 2W(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^\top)\|_{\text{nuclear}}$$

เราเปลี่ยนรูปปัญหาเพื่อให้สามารถใช้ Gradient Ascent
หาค่าตัวแปร Dual ($\boldsymbol{\Lambda}$) ได้ง่ายขึ้น

NotebookLM

ขั้นตอนการทำงานของ Manifold Muon

1. Dual Update

Run Gradient Ascent on Λ

$$\Lambda \leftarrow \Lambda + \alpha H(\Lambda)$$

2. Compute Update

Calculate direction using Matrix Sign

$$A_{\text{opt}} = -\eta \cdot \text{msign}(\ldots)$$

3. Apply Update

Step the weights

$$W \leftarrow W + A_{\text{opt}}$$

4. Retract

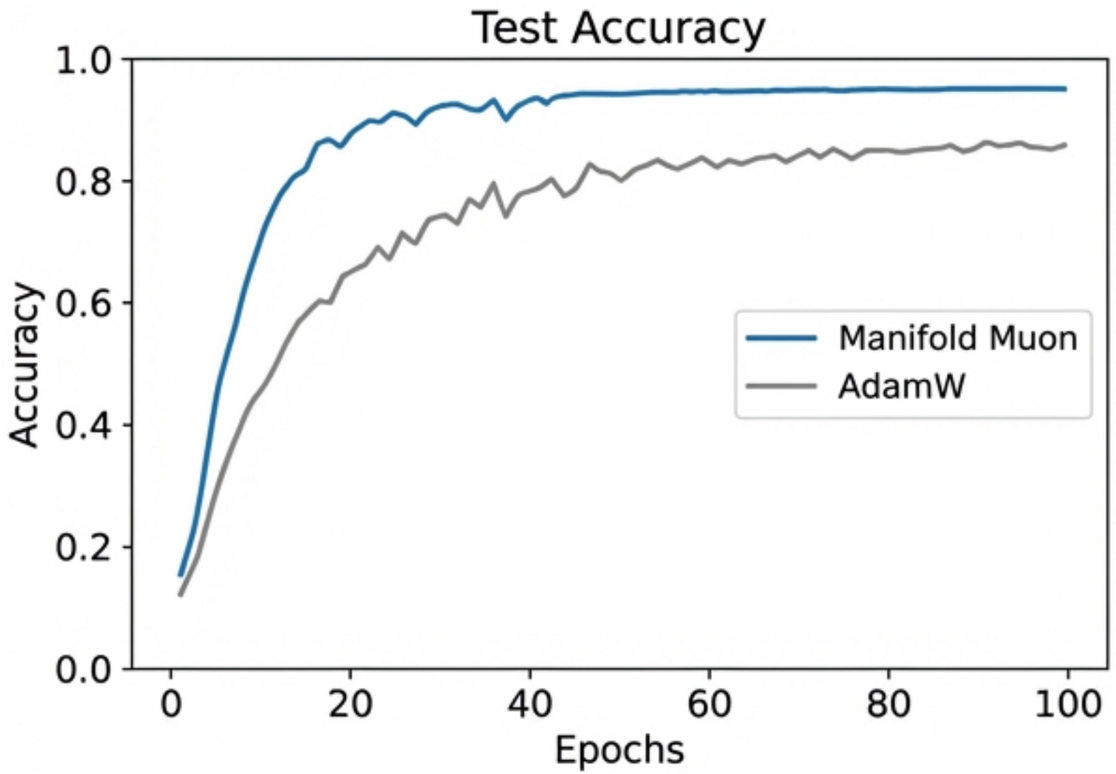
Project back to Stiefel Manifold

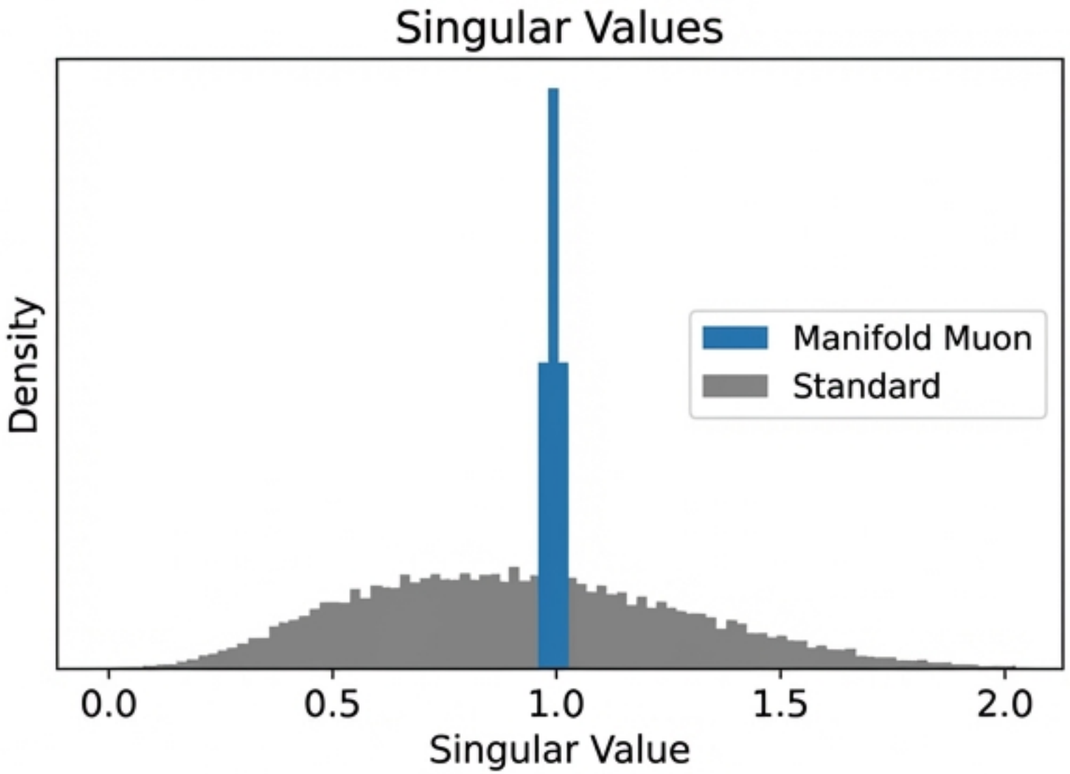
$$W \leftarrow \text{msign}(W)$$

NotebookLM

ผลลัพธ์การทดลอง: ประสิทธิภาพและความเสถียร

CIFAR-10 Training Comparison: Manifold Muon vs. AdamW



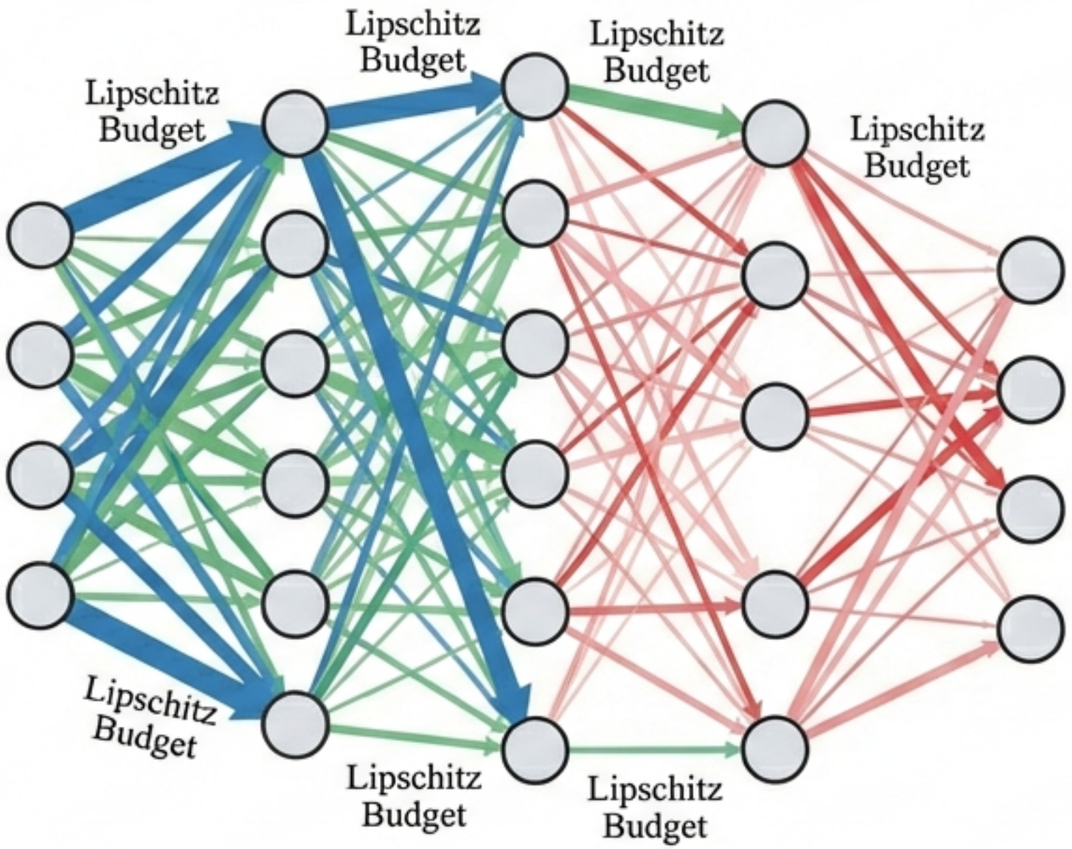


Manifold Muon ให้ความแม่นยำสูงกว่า และรักษา Singular Values ให้เป็น 1 ตลอดการเทรน

NotebookLM

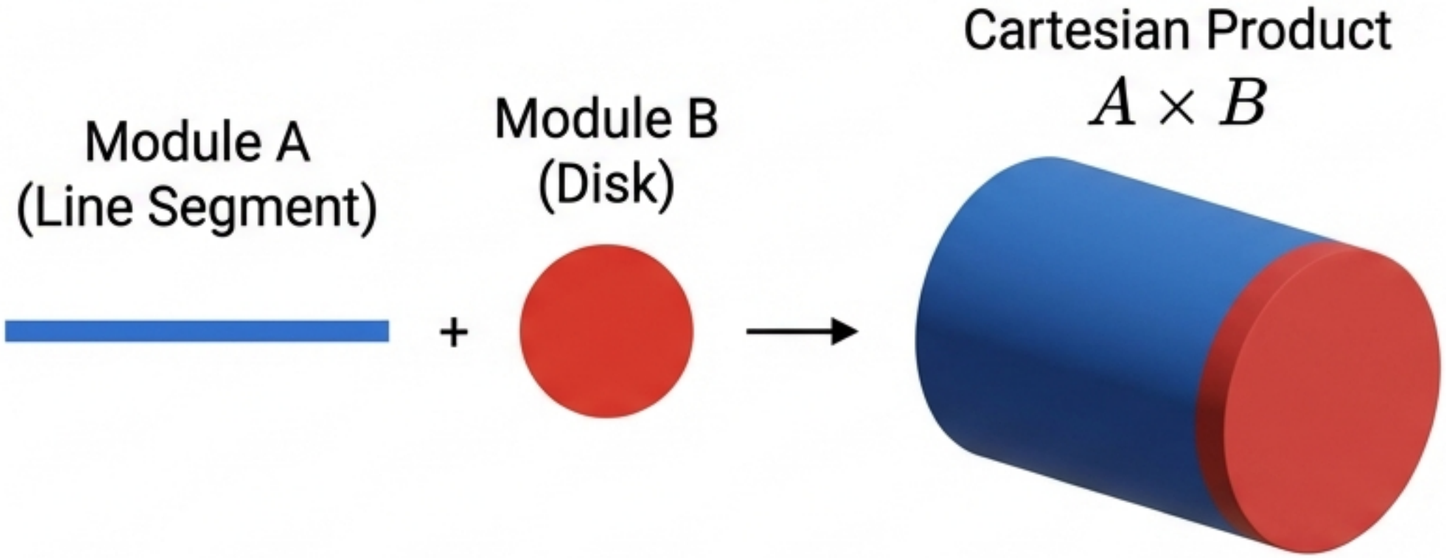
การขยายสเกลด้วย Modular Manifolds

- เราไม่สามารถมองแต่ละ Layer แยกจากกันได้
- **Budgeting Learning Rates**: เราต้องจัดสรร Learning Rate ตามความไว (Sensitivity) ของแต่ละ Layer
- Modular Manifold คือระบบที่ช่วยคำนวณความไวเหล่านี้โดยอัตโนมัติ



NotebookLM

กฎการประกอบร่างโมดูล (Rules of Composition)



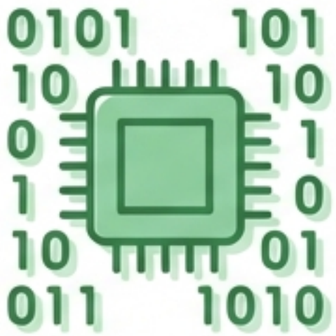
The 3 Rules
Function: $f_3 = f_2(f_1(x))$ (Composition)
Manifold: $\mathcal{M}_1 \times \mathcal{M}_2$ (Cartesian Product)
Norm: $\max(s_1\|w_1\|, s_2\|w_2\|)$ (Weighted Max)

กฎเหล่านี้ช่วยให้เราสร้าง Optimizer แยกสำหรับแต่ละ Layer ได้อย่างอิสระแต่สอดคล้องกัน

NotebookLM

ทิศทางการวิจัยในอนาคต

Attention Heads ควรอยู่บน Manifold แบบไหน?



Low-precision training & floating point constraints.

ปรับปรุง Dual Ascent ให้เร็วกว่านี้

โลกของ Neural Networks อาจไม่ใช่ Riemannian (Norm balls มีมุมแหลม)

NotebookLM

บทสรุป: สู่อนาคตของการออกแบบ Optimizer

1. **Constraint is Control:** การจำกัด Weights ให้อยู่บน Manifold ช่วยสร้างเสถียรภาพ
2. **Geometry is Key:** เรขาคณิตเป็นตัวกำหนดวิธีการอัปเดตที่ดีที่สุด (ไม่ใช่แค่ Gradient)
3. **Co-Design:** อนาคตคือการออกแบบ Architecture และ Optimizer ไปพร้อมๆ กัน

"We need to keep tensors healthy."

Thinking Machines Lab

NotebookLM