

On the Width Scaling of Neural Optimizers: A Matrix Operator Norm Perspective

Jiajin Li

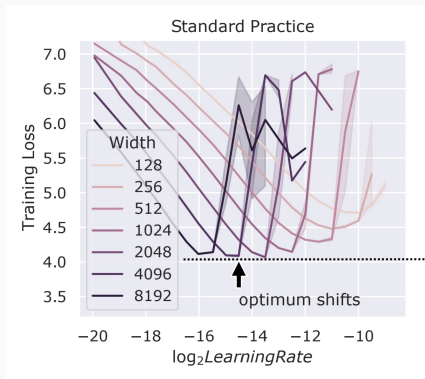
Sauder School of Business, IAM & CS

University of British Columbia

CUHK SEEM, March, 2026

Joint work with Ruihan Xu (U Chicago) and Yiping Lu (Northwestern).

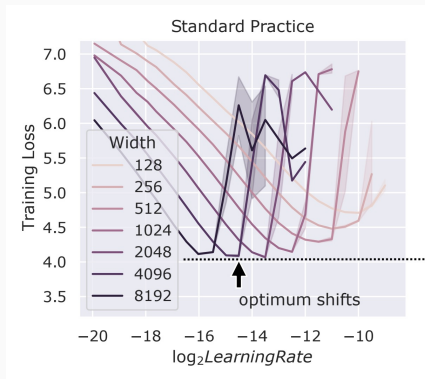
Hyperparameter Transfer [Yang et al, 2021, NeurIPS]



Key empirical observations

- The optimal learning rate **decreases** as **network width increases**.
- A learning rate tuned at width **512** can **diverge** or **slow dramatically** when scaled to width **2048**.

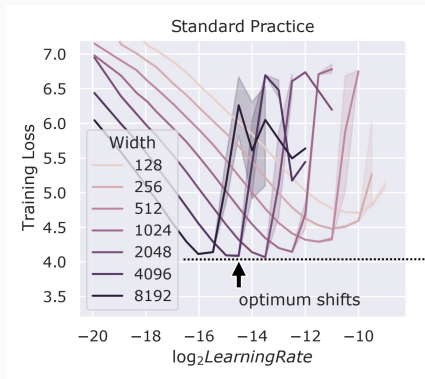
Hyperparameter Transfer [Yang et al, 2021, NeurIPS]



Key empirical observations

- The optimal learning rate **decreases** as **network width increases**.
- A learning rate tuned at width **512** can **diverge** or **slow dramatically** when scaled to width **2048**.

Hyperparameter Transfer [Yang et al, 2021, NeurIPS]

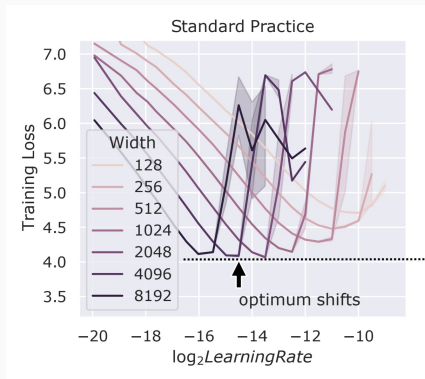


Why this matters for LLMs

Pre-training state-of-the-art base models costs $\gg$ **hundreds of millions USD**.

Hyperparameter transfer is therefore **essential**, not optional.

Hyperparameter Transfer [Yang et al, 2021, NeurIPS]

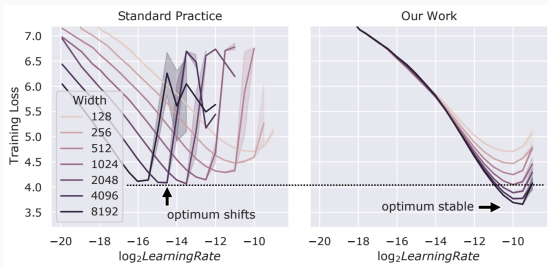


Why this matters for LLMs

Pre-training state-of-the-art base models costs $\gg$ **hundreds of millions USD**.

Hyperparameter transfer is therefore **essential**, not optional.

Hyperparameter Transfer [Yang et al, 2021, NeurIPS]



[Yang et al, 2021, NeurIPS]

Adam learning rate scaling rule:

- For Weight Matrix:

- Scale learning rate by

$$\frac{1}{d_{in}}$$

- For Embedding Matrix:

- No learning-rate scaling

Here, d_{in} denotes the effective input dimensionality of the parameter block.

What this talk is about

Design **width-independent neural optimizers** *and*
develop a principled understanding of **learning-rate scaling rules**
from a **matrix operator norm perspective**.

Problem Setup: Neural Network Optimization

We consider the constrained optimization problem

$$\min_{\mathbf{W}_{1:\ell}, \mathbf{b}_{1:\ell} \in \mathcal{C}} f(\mathbf{W}_{1:\ell}, \mathbf{b}_{1:\ell}) := \mathcal{L}(\mathbf{y}_\ell(\mathbf{x})),$$

where $\mathbf{y}_\ell(\mathbf{x}) \in \mathbb{R}$ is the output of an ℓ -layer feedforward neural network evaluated at input $\mathbf{x} \in \mathbb{R}^d$.

Network architecture

The network is defined recursively as

$$\mathbf{y}_i(\mathbf{x}) := \sigma(\mathbf{W}_i \mathbf{y}_{i-1}(\mathbf{x}) + \mathbf{b}_i), \quad i = 1, \dots, \ell,$$

with

$$\mathbf{W}_1 \in \mathbb{R}^{w \times d} \quad \text{and} \quad \mathbf{W}_i \in \mathbb{R}^{w \times w} \quad (i \geq 2).$$

Later, we write $\Theta := \{\mathbf{W}_{1:\ell}, \mathbf{b}_{1:\ell}\}$ for simplicity.

Problem Setup: Neural Network Optimization

We consider the constrained optimization problem

$$\min_{\mathbf{W}_{1:\ell}, \mathbf{b}_{1:\ell} \in \mathcal{C}} f(\mathbf{W}_{1:\ell}, \mathbf{b}_{1:\ell}) := \mathcal{L}(\mathbf{y}_\ell(\mathbf{x})),$$

where $\mathbf{y}_\ell(\mathbf{x}) \in \mathbb{R}$ is the output of an ℓ -layer feedforward neural network evaluated at input $\mathbf{x} \in \mathbb{R}^d$.

Network architecture

The network is defined recursively as

$$\mathbf{y}_i(\mathbf{x}) := \sigma(\mathbf{W}_i \mathbf{y}_{i-1}(\mathbf{x}) + \mathbf{b}_i), \quad i = 1, \dots, \ell,$$

with

$$\mathbf{W}_1 \in \mathbb{R}^{w \times d} \quad \text{and} \quad \mathbf{W}_i \in \mathbb{R}^{w \times w} \quad (i \geq 2).$$

Later, we write $\Theta := \{\mathbf{W}_{1:\ell}, \mathbf{b}_{1:\ell}\}$ for simplicity.

Decoupled Weight Decay as a Constrained Optimization Problem [Loshchilov and Hutter, 2019]

Let $\mathbf{d}^t$ denote the negative update direction at iteration t . We consider updates of the form

$$\boldsymbol{\theta}^t = \boldsymbol{\theta}^{t-1} - \eta \mathbf{d}^{t-1} - \underbrace{\eta \lambda \boldsymbol{\theta}^{t-1}}_{\text{Weight Decay}},$$

where η is the learning rate and $\lambda \in (0, 1)$ is the weight decay coefficient.

Fact

If $\|\mathbf{d}^t\| \leq \lambda R$ and $\|\boldsymbol{\theta}^0\| \leq R$, we have $\|\boldsymbol{\theta}^t\| \leq R$ for all $t \geq 0$.

View decoupled weight decay as enforcing an implicit norm ball constraint !

Decoupled Weight Decay as a Constrained Optimization Problem [Loshchilov and Hutter, 2019]

Let $\mathbf{d}^t$ denote the negative update direction at iteration t . We consider updates of the form

$$\boldsymbol{\theta}^t = \boldsymbol{\theta}^{t-1} - \eta \mathbf{d}^{t-1} - \underbrace{\eta \lambda \boldsymbol{\theta}^{t-1}}_{\text{Weight Decay}},$$

where η is the learning rate and $\lambda \in (0, 1)$ is the weight decay coefficient.

Fact

If $\|\mathbf{d}^t\| \leq \lambda R$ and $\|\boldsymbol{\theta}^0\| \leq R$, we have $\|\boldsymbol{\theta}^t\| \leq R$ for all $t \geq 0$.

View decoupled weight decay as enforcing an implicit norm ball constraint !

Decoupled Weight Decay as a Constrained Optimization Problem [Loshchilov and Hutter, 2019]

Let $\mathbf{d}^t$ denote the negative update direction at iteration t . We consider updates of the form

$$\boldsymbol{\theta}^t = \boldsymbol{\theta}^{t-1} - \eta \mathbf{d}^{t-1} - \underbrace{\eta \lambda \boldsymbol{\theta}^{t-1}}_{\text{Weight Decay}},$$

where η is the learning rate and $\lambda \in (0, 1)$ is the weight decay coefficient.

Fact

If $\|\mathbf{d}^t\| \leq \lambda R$ and $\|\boldsymbol{\theta}^0\| \leq R$, we have $\|\boldsymbol{\theta}^t\| \leq R$ for all $t \geq 0$.

View decoupled weight decay as enforcing an implicit norm ball constraint !

A Brief Review of Successful Optimizers — AdamW, signSGD

- AdamW [Kingma and Ba, 2014] was regarded as the gold standard optimizer for large-scale neural network training.

$$\mathbf{g}^t = \nabla f(\boldsymbol{\theta}^{t-1}, \xi_t) \quad (\text{compute gradient})$$

$$\mathbf{m}^t = \beta_1 \mathbf{m}^{t-1} + (1 - \beta_1) \mathbf{g}^t \quad (\text{estimate first order moment})$$

$$\mathbf{v}^t = \beta_2 \mathbf{v}^{t-1} + (1 - \beta_2) \mathbf{g}^t \odot \mathbf{g}^t \quad (\text{estimate second order moment})$$

$$\hat{\mathbf{m}}^t = \frac{\mathbf{m}^t}{1 - \beta_1}, \hat{\mathbf{v}}^t = \frac{\mathbf{v}^t}{1 - \beta_2} \quad (\text{bias correction})$$

$$\boldsymbol{\theta}^t = \boldsymbol{\theta}^{t-1} - \eta \left(\frac{\hat{\mathbf{m}}^t}{\sqrt{\hat{\mathbf{v}}^t} + \epsilon} \right) - \eta \lambda \boldsymbol{\theta}^{t-1}$$

- Adam approximate signSGD [Bernstein et al, 2018, ICML].

$$\boldsymbol{\theta}^t \approx \boldsymbol{\theta}^{t-1} - \eta \frac{\mathbf{g}^t}{\sqrt{\mathbf{g}^t \odot \mathbf{g}^t}} \approx \boldsymbol{\theta}^{t-1} - \eta \cdot \text{sign}(\mathbf{g}^t).$$

Key idea

Lion can be viewed as **SignSGD** + **momentum** + **decoupled weight decay**.

$$\mathbf{g}^t = \nabla f(\boldsymbol{\theta}^{t-1}, \xi_t)$$

$$\mathbf{m}^t = \beta_1 \mathbf{m}^{t-1} + (1 - \beta_1) \mathbf{g}^t$$

$$\boldsymbol{\theta}^t = \boldsymbol{\theta}^{t-1} - \eta \text{sign}(\mathbf{m}^t) - \eta \lambda \boldsymbol{\theta}^{t-1}$$

- Automatic discovering optimizers through RL/meta-learning
- Less memory compared with AdamW

A Brief Review of Successful Optimizers — Muon

Key idea

Change the optimization geometry from a **vector space** to a **matrix space**.

$$\mathbf{G}^t = \nabla f(\Theta^{t-1}, \xi_t)$$

$$\mathbf{M}^t = \beta_1 \mathbf{M}^{t-1} + (1 - \beta_1) \mathbf{G}^t$$

$$\Theta^t = \Theta^{t-1} - \eta \text{matrixsign}(\mathbf{M}^t) - \eta \lambda \Theta^{t-1}$$

- $\text{matrixsign}(\mathbf{U}\Sigma\mathbf{V}^\top) = \mathbf{U}\mathbf{V}^\top$ — can be approximated via **Newton-Schultz Iterations**

[Jordan et al, 2024]

Two Controversial Learning Rate Scaling Rules for Muon

- μ P version [Yang et al., 2023]:

$$\sqrt{\frac{d_{\text{out}}}{d_{\text{in}}}} \cdot \underbrace{\eta_{\text{base}}}_{\text{optimal LR for the base (small) model}}$$

- Grafting in spectral geometry
- Moonlight version [Liu et al., 2025, Moonshot AI]:

$$\sqrt{\max\{d_{\text{out}}, d_{\text{in}}\}} \cdot \underbrace{\eta_{\text{Adam}}}_{\text{tuned LR using AdamW}}$$

- Deployed in Kimi
- Grafting in Euclidean geometry [Agarwal et al, 2020]

Here, d_{out} denotes the effective output dimensionality of the parameter block.

Unifying via Steepest Descent Under Different Matrix Operator Norms

Matrix Thinking

All these optimizers admit a unified interpretation as steepest descent under different matrix operator norms.

$$D^t = \arg \min_{\|D\|_F \leq 1} \langle M^t, D \rangle \quad (*)$$

Definition (Operator Norm)

Let $\|\cdot\|_{\text{in}}$ and $\|\cdot\|_{\text{out}}$ be norms on $\mathbb{R}^n$ and $\mathbb{R}^m$. The operator norm of D induced by these norms is defined as

$$\|D\|_{\text{in} \rightarrow \text{out}} := \sup_{\|x\|_{\text{in}}=1} \|Dx\|_{\text{out}}.$$

Unifying via Steepest Descent Under Different Matrix Operator Norms

Matrix Thinking

All these optimizers admit a unified interpretation as steepest descent under different matrix operator norms.

$$D^t = \arg \min_{\|D\|_? \leq 1} \langle M^t, D \rangle \quad (*)$$

Definition (Operator Norm)

Let $\|\cdot\|_{\text{in}}$ and $\|\cdot\|_{\text{out}}$ be norms on $\mathbb{R}^n$ and $\mathbb{R}^m$. The operator norm of D induced by these norms is defined as

$$\|D\|_{\text{in} \rightarrow \text{out}} := \sup_{\|x\|_{\text{in}}=1} \|Dx\|_{\text{out}}.$$

Unifying via Steepest Descent Under Different Matrix Operator Norms

Matrix Thinking

All these optimizers admit a unified interpretation as steepest descent under different matrix operator norms.

$$D^t = \arg \min_{\|D\|_{\gamma} \leq 1} \langle M^t, D \rangle \quad (*)$$

Definition (Operator Norm)

Let $\|\cdot\|_{\text{in}}$ and $\|\cdot\|_{\text{out}}$ be norms on $\mathbb{R}^n$ and $\mathbb{R}^m$. The operator norm of D induced by these norms is defined as

$$\|D\|_{\text{in} \rightarrow \text{out}} := \sup_{\|x\|_{\text{in}}=1} \|Dx\|_{\text{out}}.$$

Select the Correct Matrix Operator Norm

Example (SignSGD/Adam, $\ell_1 \rightarrow \ell_\infty$)

The element-wise ℓ_∞ norm of $\mathbf{D}$ coincides with its $\ell_1 \rightarrow \ell_\infty$ operator norm, i.e.,

$$\|\mathbf{D}\|_{1 \rightarrow \infty} = \max_{i,j} |\mathbf{D}_{i,j}| \quad \text{and} \quad \mathbf{D}^* = \arg \min_{\|\mathbf{D}\|_{1 \rightarrow \infty} \leq 1} \langle \mathbf{M}, \mathbf{D} \rangle = -\text{sign}(\mathbf{M}).$$

Example (Muon, $\ell_2 \rightarrow \ell_2$)

The spectral norm of $\mathbf{D}$ coincides with its $\ell_2 \rightarrow \ell_2$ operator norm, i.e.,

$$\|\mathbf{D}\|_{2 \rightarrow 2} = \|\mathbf{D}\|_2 \quad \text{and} \quad \mathbf{D}^* = \arg \min_{\|\mathbf{D}\|_{2 \rightarrow 2} \leq 1} \langle \mathbf{M}, \mathbf{D} \rangle = -\text{matrixsign}(\mathbf{M}).$$

Select the Correct Matrix Operator Norm

Example (SignSGD/Adam, $\ell_1 \rightarrow \ell_\infty$)

The element-wise ℓ_∞ norm of $\mathbf{D}$ coincides with its $\ell_1 \rightarrow \ell_\infty$ operator norm, i.e.,

$$\|\mathbf{D}\|_{1 \rightarrow \infty} = \max_{i,j} |\mathbf{D}_{i,j}| \quad \text{and} \quad \mathbf{D}^* = \arg \min_{\|\mathbf{D}\|_{1 \rightarrow \infty} \leq 1} \langle \mathbf{M}, \mathbf{D} \rangle = -\text{sign}(\mathbf{M}).$$

Example (Muon, $\ell_2 \rightarrow \ell_2$)

The spectral norm of $\mathbf{D}$ coincides with its $\ell_2 \rightarrow \ell_2$ operator norm, i.e.,

$$\|\mathbf{D}\|_{2 \rightarrow 2} = \|\mathbf{D}\|_2 \quad \text{and} \quad \mathbf{D}^* = \arg \min_{\|\mathbf{D}\|_{2 \rightarrow 2} \leq 1} \langle \mathbf{M}, \mathbf{D} \rangle = -\text{matrixsign}(\mathbf{M}).$$

Select the Correct Matrix Operator Norm II [Bernstein and Newhouse, 2024]

Example (Column Normalization, $\ell_1 \rightarrow \ell_q$)

$$\|\mathbf{D}\|_{1 \rightarrow q} = \max_{c \in [n]} \|\mathbf{D}_{:,c}\|_q, \text{ and}$$

$$\mathbf{D}^* = \text{colnorm}_q(\mathbf{M}) \quad \text{and} \quad \text{colnorm}_q(\mathbf{M})_{:,c} := \frac{\text{sign}(\mathbf{M}_{:,c}) \odot |\mathbf{M}_{:,c}|^{q^*-1}}{\|\mathbf{M}_{:,c}\|_{q^*}^{q^*-1}},$$

where $\frac{1}{q} + \frac{1}{q^*} = 1$.

Example (Row Normalization, $\ell_p \rightarrow \ell_\infty$)

$$\|\mathbf{D}\|_{p \rightarrow \infty} = \max_{r \in [m]} \|\mathbf{D}_{r,:}\|_{p^*},$$

$$\mathbf{D}^* = \text{rownorm}_p(\mathbf{M}) \quad \text{and} \quad \text{rownorm}_p(\mathbf{M})_{r,:} := \frac{\text{sign}(\mathbf{M}_{r,:}) \odot |\mathbf{M}_{r,:}|^{p-1}}{\|\mathbf{M}_{r,:}\|_p^{p-1}},$$

where $\frac{1}{p} + \frac{1}{p^*} = 1$.

Select the Correct Matrix Operator Norm II [Bernstein and Newhouse, 2024]

Example (Column Normalization, $\ell_1 \rightarrow \ell_q$)

$$\|\mathbf{D}\|_{1 \rightarrow q} = \max_{c \in [n]} \|\mathbf{D}_{:,c}\|_q, \text{ and}$$

$$\mathbf{D}^* = \text{colnorm}_q(\mathbf{M}) \quad \text{and} \quad \text{colnorm}_q(\mathbf{M})_{:,c} := \frac{\text{sign}(\mathbf{M}_{:,c}) \odot |\mathbf{M}_{:,c}|^{q^*-1}}{\|\mathbf{M}_{:,c}\|_{q^*}^{q^*-1}},$$

where $\frac{1}{q} + \frac{1}{q^*} = 1$.

Example (Row Normalization, $\ell_p \rightarrow \ell_\infty$)

$$\|\mathbf{D}\|_{p \rightarrow \infty} = \max_{r \in [m]} \|\mathbf{D}_{r,:}\|_{p^*},$$

$$\mathbf{D}^* = \text{rownorm}_p(\mathbf{M}) \quad \text{and} \quad \text{rownorm}_p(\mathbf{M})_{r,:} := \frac{\text{sign}(\mathbf{M}_{r,:}) \odot |\mathbf{M}_{r,:}|^{p-1}}{\|\mathbf{M}_{r,:}\|_p^{p-1}},$$

where $\frac{1}{p} + \frac{1}{p^*} = 1$.

SignSGD/AdamW	Column Normalization	Row Normalization	Muon
$\ \cdot\ _{1 \rightarrow \infty}$	$\ \cdot\ _{1 \rightarrow q}$	$\ \cdot\ _{p \rightarrow \infty}$	$\ \cdot\ _{2 \rightarrow 2}$

Choosing the right matrix operator norm is the key to designing new optimizers!

SignSGD/AdamW	Column Normalization	Row Normalization	Muon
$\ \cdot\ _{1 \rightarrow \infty}$	$\ \cdot\ _{1 \rightarrow q}$	$\ \cdot\ _{p \rightarrow \infty}$	$\ \cdot\ _{2 \rightarrow 2}$

Choosing the right matrix operator norm is the key to designing new optimizers!

Geometry-Aware Principles for Optimizer Design

- **Low per-iteration computational cost**

- Choosing matrix operator geometries $\|\cdot\|_{p \rightarrow q}$ with $p \leq q$.

- **Neural network landscape**

- **Width-independent Lipschitz bound:** the network's *forward sensitivity* is bounded independently of width.
- **Width-independent smoothness bound:** the *second-order sensitivity* is bounded independently of width.

These width-independent bounds naturally induce *learning rate scaling rules* that are stable across widths, enabling effective hyperparameter transfer.

Geometry-Aware Principles for Optimizer Design

- **Low per-iteration computational cost**

- Choosing matrix operator geometries $\|\cdot\|_{p \rightarrow q}$ with $p \leq q$.

- **Neural network landscape**

- **Width-independent Lipschitz bound:** the network's *forward sensitivity* is bounded independently of width.
- **Width-independent smoothness bound:** the *second-order sensitivity* is bounded independently of width.

These width-independent bounds naturally induce ***learning rate scaling rules*** that are stable across widths, enabling effective hyperparameter transfer.

Width-independent Lipschitz Continuous NN

Observation

Standard operator norms ($p \rightarrow q$) fail to yield width-independent Lipschitz bounds.

- Each linear map $\mathbf{W}_\ell$ sends features from one geometric space to the next pre-activation space. The operator norm of $\mathbf{W}_\ell$ induces a layer-wise propagation of stability, i.e.,

$$\|\nabla_{\mathbf{w}_i} \mathbf{y}_{l+1}\|_{\text{out}} \leq \|\mathbf{W}_l \nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{out}} \leq \|\mathbf{W}_l\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{in}}, \forall i < \ell.$$

- Cross-layer stability bound using matrix operator norms requires: $\|\cdot\|_{\text{in}} \leq \|\cdot\|_{\text{out}}$

$$\|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{out}} \leq \|\mathbf{W}_\ell\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_\ell\|_{\text{in}}$$

$$\|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+2}\|_{\text{out}} \leq \|\mathbf{W}_{\ell+1}\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{in}}$$

$$\Rightarrow \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+2}\|_{\text{out}} \leq \prod_{k=l}^{l+1} \|\mathbf{W}_k\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_\ell\|_{\text{in}}$$

Width-independent Lipschitz Continuous NN

Observation

Standard operator norms ($p \rightarrow q$) fail to yield width-independent Lipschitz bounds.

- Each linear map $\mathbf{W}_\ell$ sends features from one geometric space to the next pre-activation space. The operator norm of $\mathbf{W}_\ell$ induces a layer-wise propagation of stability, i.e.,

$$\|\nabla_{\mathbf{w}_i} \mathbf{y}_{l+1}\|_{\text{out}} \leq \|\mathbf{W}_l \nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{out}} \leq \|\mathbf{W}_l\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{in}}, \forall i < \ell.$$

- Cross-layer stability bound using matrix operator norms requires: $\|\cdot\|_{\text{in}} \leq \|\cdot\|_{\text{out}}$

$$\begin{aligned} \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{out}} &\leq \|\mathbf{W}_\ell\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_\ell\|_{\text{in}} \\ \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+2}\|_{\text{out}} &\leq \|\mathbf{W}_{\ell+1}\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{in}} \end{aligned}$$

$$\Rightarrow \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+2}\|_{\text{out}} \leq \prod_{k=\ell}^{\ell+1} \|\mathbf{W}_k\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_\ell\|_{\text{in}}$$

Width-independent Lipschitz Continuous NN

Observation

Standard operator norms ($p \rightarrow q$) fail to yield width-independent Lipschitz bounds.

- Each linear map $\mathbf{W}_\ell$ sends features from one geometric space to the next pre-activation space. The operator norm of $\mathbf{W}_\ell$ induces a layer-wise propagation of stability, i.e.,

$$\|\nabla_{\mathbf{w}_i} \mathbf{y}_{l+1}\|_{\text{out}} \leq \|\mathbf{W}_l \nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{out}} \leq \|\mathbf{W}_l\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{in}}, \forall i < \ell.$$

- Cross-layer stability bound using matrix operator norms requires: $\|\cdot\|_{\text{in}} \leq \|\cdot\|_{\text{out}}$

$$\begin{aligned} \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{out}} &\leq \|\mathbf{W}_\ell\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_\ell\|_{\text{in}} \\ \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+2}\|_{\text{out}} &\leq \|\mathbf{W}_{\ell+1}\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{in}} \end{aligned}$$

$$\Rightarrow \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+2}\|_{\text{out}} \leq \prod_{k=l}^{l+1} \|\mathbf{W}_k\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_\ell\|_{\text{in}}$$

Width-independent Lipschitz Continuous NN

Observation

Standard operator norms ($p \rightarrow q$) fail to yield width-independent Lipschitz bounds.

- Each linear map $\mathbf{W}_\ell$ sends features from one geometric space to the next pre-activation space. The operator norm of $\mathbf{W}_\ell$ induces a layer-wise propagation of stability, i.e.,

$$\|\nabla_{\mathbf{w}_i} \mathbf{y}_{l+1}\|_{\text{out}} \leq \|\mathbf{W}_l \nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{out}} \leq \|\mathbf{W}_l\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{in}}, \forall i < \ell.$$

- Cross-layer stability bound using matrix operator norms requires: $\|\cdot\|_{\text{in}} \leq \|\cdot\|_{\text{out}}$

$$\|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{out}} \leq \|\mathbf{W}_\ell\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_\ell\|_{\text{in}}$$

$$\|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+2}\|_{\text{out}} \leq \|\mathbf{W}_{\ell+1}\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{in}}$$

Example

Consider $\mathbf{x} = (1, 1, \dots, 1) \in \mathbb{R}^n$. We have $\|\mathbf{x}\|_\infty = 1 < \|\mathbf{x}\|_p = n^{1/p} < \|\mathbf{x}\|_1 = n$.

Width-independent Lipschitz Continuous NN

Observation

Standard operator norms ($p \rightarrow q$) fail to yield width-independent Lipschitz bounds.

- Each linear map $\mathbf{W}_\ell$ sends features from one geometric space to the next pre-activation space. The operator norm of $\mathbf{W}_\ell$ induces a layer-wise propagation of stability, i.e.,

$$\|\nabla_{\mathbf{w}_i} \mathbf{y}_{l+1}\|_{\text{out}} \leq \|\mathbf{W}_l \nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{out}} \leq \|\mathbf{W}_l\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{in}}, \forall i < \ell.$$

- Cross-layer stability bound using matrix operator norms requires: $\|\cdot\|_{\text{in}} \leq \|\cdot\|_{\text{out}}$

$$\begin{aligned} \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{out}} &\leq \|\mathbf{W}_\ell\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_\ell\|_{\text{in}} \\ \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+2}\|_{\text{out}} &\leq \|\mathbf{W}_{\ell+1}\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{in}} \end{aligned}$$

Example

Consider $\mathbf{x} = (1, 1, \dots, 1) \in \mathbb{R}^n$. We have $\|\mathbf{x}\|_\infty = 1 < \|\mathbf{x}\|_p = n^{1/p} < \|\mathbf{x}\|_1 = n$.

Width-independent Lipschitz Continuous NN

Observation

Standard operator norms ($p \rightarrow q$) fail to yield width-independent Lipschitz bounds.

- Each linear map $\mathbf{W}_\ell$ sends features from one geometric space to the next pre-activation space. The operator norm of $\mathbf{W}_\ell$ induces a layer-wise propagation of stability, i.e.,

$$\|\nabla_{\mathbf{w}_i} \mathbf{y}_{l+1}\|_{\text{out}} \leq \|\mathbf{W}_l \nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{out}} \leq \|\mathbf{W}_l\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_l\|_{\text{in}}, \forall i < \ell.$$

- Cross-layer stability bound using matrix operator norms requires: $\|\cdot\|_{\text{in}} \leq \|\cdot\|_{\text{out}}$

$$\begin{aligned} \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{out}} &\leq \|\mathbf{W}_\ell\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_\ell\|_{\text{in}} \\ \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+2}\|_{\text{out}} &\leq \|\mathbf{W}_{\ell+1}\|_{\text{in} \rightarrow \text{out}} \|\nabla_{\mathbf{w}_i} \mathbf{y}_{\ell+1}\|_{\text{in}} \end{aligned}$$

Example

Consider $\mathbf{x} = (1, 1, \dots, 1) \in \mathbb{R}^n$. We have $\|\mathbf{x}\|_\infty = 1 < \|\mathbf{x}\|_p = n^{1/p} < \|\mathbf{x}\|_1 = n$.

How to make the matrix operator norm play nicely together?

Definition (Mean-Normalized Norm)

We introduce a new, width-aware geometry via the mean-normalized norms $\|\cdot\|_{(p,\text{mean})}$:

$$\|\mathbf{x}\|_{(p,\text{mean})} := \left(\frac{1}{n} \sum_{i=1}^n |\mathbf{x}_i|^p \right)^{1/p} = n^{-1/p} \|\mathbf{x}\|_p.$$

- The factor $n^{-1/p}$ precisely cancels the dimensional scaling of the ℓ_p embeddings.

Fact

Let $\mathbf{x} \in \mathbb{R}^n$ and $1 \leq p \leq q \leq \infty$, then $\|\mathbf{x}\|_{(p,\text{mean})} \leq \|\mathbf{x}\|_{(q,\text{mean})}$.

- Choosing the matrix operator norm $\|\cdot\|_{(p,\text{mean}) \rightarrow (q,\text{mean})}$ with $p \leq q$!

How to make the matrix operator norm play nicely together?

Definition (Mean-Normalized Norm)

We introduce a new, width-aware geometry via the mean-normalized norms $\|\cdot\|_{(p,\text{mean})}$:

$$\|\mathbf{x}\|_{(p,\text{mean})} := \left(\frac{1}{n} \sum_{i=1}^n |\mathbf{x}_i|^p \right)^{1/p} = n^{-1/p} \|\mathbf{x}\|_p.$$

- The factor $n^{-1/p}$ precisely cancels the dimensional scaling of the ℓ_p embeddings.

Fact

Let $\mathbf{x} \in \mathbb{R}^n$ and $1 \leq p \leq q \leq \infty$, then $\|\mathbf{x}\|_{(p,\text{mean})} \leq \|\mathbf{x}\|_{(q,\text{mean})}$.

- Choosing the matrix operator norm $\|\cdot\|_{(p,\text{mean}) \rightarrow (q,\text{mean})}$ with $p \leq q$!

How to make the matrix operator norm play nicely together?

Definition (Mean-Normalized Norm)

We introduce a new, width-aware geometry via the mean-normalized norms $\|\cdot\|_{(p,\text{mean})}$:

$$\|\mathbf{x}\|_{(p,\text{mean})} := \left(\frac{1}{n} \sum_{i=1}^n |\mathbf{x}_i|^p \right)^{1/p} = n^{-1/p} \|\mathbf{x}\|_p.$$

- The factor $n^{-1/p}$ precisely cancels the dimensional scaling of the ℓ_p embeddings.

Fact

Let $\mathbf{x} \in \mathbb{R}^n$ and $1 \leq p \leq q \leq \infty$, then $\|\mathbf{x}\|_{(p,\text{mean})} \leq \|\mathbf{x}\|_{(q,\text{mean})}$.

- Choosing the matrix operator norm $\|\cdot\|_{(p,\text{mean}) \rightarrow (q,\text{mean})}$ with $p \leq q$!

Width-independent Lipschitz Bound

Theorem (Width-Independent Lipschitz Bound Under $(p, \text{mean}) \rightarrow (q, \text{mean})$ Geometry)

Suppose that (1) $1 \leq p \leq q < \infty$; (2) the input is bounded; (3) both σ and $\mathcal{L}$ have bounded first derivatives. Define the parameter set as (4) $\Omega_C := \{\Theta : \|\Theta\|_{\text{block}} \leq C\}$, where the block norm $\|\Theta\|_{\text{block}}$ is defined as

$$\max \left\{ \|\mathbf{W}_1\|_{1 \rightarrow (q, \text{mean})}, \max_{2 \leq i \leq K-1} \|\mathbf{W}_i\|_{(p, \text{mean}) \rightarrow (q, \text{mean})}, \|\mathbf{W}_K\|_{(p, \text{mean}) \rightarrow \infty}, \max_i \|\mathbf{b}_i\|_{\infty} \right\}.$$

Then the loss function is M -Lipschitz continuous on Ω_C , where M is independent of the width.

Width-independent L -Smooth NN

Definition (L -Smoothness)

Let $f : \Omega \rightarrow \mathbb{R}$ be differentiable on a convex set Ω equipped with a norm $\|\cdot\|$. We say that f is L -smooth with respect to $\|\cdot\|$ if, for all $\mathbf{Z}, \mathbf{Z}' \in \Omega$,

$$\|\nabla f(\mathbf{Z}) - \nabla f(\mathbf{Z}')\|_* \leq L \|\mathbf{Z} - \mathbf{Z}'\|,$$

where $\|\cdot\|_*$ denotes the dual norm of $\|\cdot\|$.

Theorem (Smoothness Bound Under $(p, \text{mean}) \rightarrow (q, \text{mean})$ Geometry)

Suppose that (1) $1 \leq p \leq q < \infty$; (2) the input is bounded; (3) both σ and $\mathcal{L}$ have bounded second derivatives. Then, we have

$$\|\nabla f(\Theta) - \nabla f(\Theta')\|_{\text{block},*} \leq L w^{\max\left(0, \frac{2}{q} - \frac{1}{p}\right)} \|\Theta - \Theta'\|_{\text{block}} \quad \forall \Theta, \Theta' \in \Omega_C,$$

where $L > 0$ is independent of the width.

$$q \geq 2p \quad !!!$$

Width-independent L -Smooth NN

Definition (L -Smoothness)

Let $f : \Omega \rightarrow \mathbb{R}$ be differentiable on a convex set Ω equipped with a norm $\|\cdot\|$. We say that f is L -smooth with respect to $\|\cdot\|$ if, for all $\mathbf{Z}, \mathbf{Z}' \in \Omega$,

$$\|\nabla f(\mathbf{Z}) - \nabla f(\mathbf{Z}')\|_* \leq L \|\mathbf{Z} - \mathbf{Z}'\|,$$

where $\|\cdot\|_*$ denotes the dual norm of $\|\cdot\|$.

Theorem (Smoothness Bound Under $(p, \text{mean}) \rightarrow (q, \text{mean})$ Geometry)

Suppose that (1) $1 \leq p \leq q < \infty$; (2) the input is bounded; (3) both σ and $\mathcal{L}$ have bounded second derivatives. Then, we have

$$\|\nabla f(\Theta) - \nabla f(\Theta')\|_{\text{block},*} \leq L w^{\max\left(0, \frac{2}{q} - \frac{1}{p}\right)} \|\Theta - \Theta'\|_{\text{block}} \quad \forall \Theta, \Theta' \in \Omega_C,$$

where $L > 0$ is independent of the width.

$$q \geq 2p \quad !!!$$

Width-independent L -Smooth NN

Definition (L -Smoothness)

Let $f : \Omega \rightarrow \mathbb{R}$ be differentiable on a convex set Ω equipped with a norm $\|\cdot\|$. We say that f is L -smooth with respect to $\|\cdot\|$ if, for all $\mathbf{Z}, \mathbf{Z}' \in \Omega$,

$$\|\nabla f(\mathbf{Z}) - \nabla f(\mathbf{Z}')\|_* \leq L \|\mathbf{Z} - \mathbf{Z}'\|,$$

where $\|\cdot\|_*$ denotes the dual norm of $\|\cdot\|$.

Theorem (Smoothness Bound Under $(p, \text{mean}) \rightarrow (q, \text{mean})$ Geometry)

Suppose that (1) $1 \leq p \leq q < \infty$; (2) the input is bounded; (3) both σ and $\mathcal{L}$ have bounded second derivatives. Then, we have

$$\|\nabla f(\Theta) - \nabla f(\Theta')\|_{\text{block},*} \leq L w^{\max\left(0, \frac{2}{q} - \frac{1}{p}\right)} \|\Theta - \Theta'\|_{\text{block}} \quad \forall \Theta, \Theta' \in \Omega_C,$$

where $L > 0$ is independent of the width.

$$q \geq 2p \quad !!!$$

Lemma (Hölder-type Inequality for Hadamard Product)

Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. For any $p, q \geq 1$, we have

$$\|\mathbf{x} \odot \mathbf{y}\|_{(p, \text{mean})} \leq n^{\max\left(0, \frac{2}{q} - \frac{1}{p}\right)} \|\mathbf{x}\|_{(q, \text{mean})} \|\mathbf{y}\|_{(q, \text{mean})}.$$

The directional Hessian can be reduced to controlling this quadratic term:

$$\begin{aligned} |\nabla^2 f(\mathbf{W})[\Delta \mathbf{W}^1, \Delta \mathbf{W}^2]| &\lesssim \|(\Delta \mathbf{W}^1 \mathbf{x}) \odot (\Delta \mathbf{W}^2 \mathbf{x})\|_{(p, \text{mean})} \\ &\lesssim n^{\max\left(0, \frac{2}{q} - \frac{1}{p}\right)} \|\Delta \mathbf{W}^1 \mathbf{x}\|_{(q, \text{mean})} \|\Delta \mathbf{W}^2 \mathbf{x}\|_{(q, \text{mean})}. \end{aligned}$$

Due to the definition of the matrix operator norm, we have

$$\|\Delta \mathbf{W}^1 \mathbf{x}\|_{(q, \text{mean})} \leq \|\Delta \mathbf{W}^1\|_{(p, \text{mean}) \rightarrow (q, \text{mean})} \|\mathbf{x}\|_{(p, \text{mean})} \leq \|\mathbf{x}\|_{(p, \text{mean})}.$$

Lemma (Hölder-type Inequality for Hadamard Product)

Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. For any $p, q \geq 1$, we have

$$\|\mathbf{x} \odot \mathbf{y}\|_{(p, \text{mean})} \leq n^{\max\left(0, \frac{2}{q} - \frac{1}{p}\right)} \|\mathbf{x}\|_{(q, \text{mean})} \|\mathbf{y}\|_{(q, \text{mean})}.$$

The directional Hessian can be reduced to controlling this quadratic term:

$$\begin{aligned} |\nabla^2 f(\mathbf{W})[\Delta \mathbf{W}^1, \Delta \mathbf{W}^2]| &\lesssim \|(\Delta \mathbf{W}^1 \mathbf{x}) \odot (\Delta \mathbf{W}^2 \mathbf{x})\|_{(p, \text{mean})} \\ &\lesssim n^{\max\left(0, \frac{2}{q} - \frac{1}{p}\right)} \|\Delta \mathbf{W}^1 \mathbf{x}\|_{(q, \text{mean})} \|\Delta \mathbf{W}^2 \mathbf{x}\|_{(q, \text{mean})}. \end{aligned}$$

Due to the definition of the matrix operator norm, we have

$$\|\Delta \mathbf{W}^1 \mathbf{x}\|_{(q, \text{mean})} \leq \|\Delta \mathbf{W}^1\|_{(p, \text{mean}) \rightarrow (q, \text{mean})} \|\mathbf{x}\|_{(p, \text{mean})} \leq \|\mathbf{x}\|_{(p, \text{mean})}.$$

Width Aware Learning Rate Scaling Rule

Matrix Operator Geometry Aware (MOGA) Scaling

For a matrix $D \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ and $1 \leq p \leq q \leq \infty$, we have

$$\|D\|_{(p, \text{mean}) \rightarrow (q, \text{mean})} = \frac{d_{\text{in}}^{1/p}}{d_{\text{out}}^{1/q}} \|D\|_{p \rightarrow q}.$$

- In the case of Adam ($\ell_1 \rightarrow \ell_\infty$), MOGA scaling exactly recovers the μP scaling $\frac{1}{d_{\text{in}}}$.
- For Muon ($\ell_2 \rightarrow \ell_2$), MOGA scaling recovers the μP versioned Muon $\sqrt{\frac{d_{\text{out}}}{d_{\text{in}}}}$.
 - [Yang et al, 2023] proposed a **spectral condition** requiring the update direction to have spectral norm $\sqrt{d_{\text{out}}/d_{\text{in}}}$.
 - However, when $p = q = 2$, we have $L = \mathcal{O}(\sqrt{w})$, which matches the learning rate scaling rule in the **Moonlight** version.

Matrix Operator Geometry Aware (MOGA) Optimizer

Algorithm 1 MOGA: Matrix-Operator-Geometry-Aware Steepest Descent

Require: Learning rates $\{\eta^t\}_{t \geq 0}$, momentum parameters $\beta_1, \beta_2 \in (0, 1)$, initial parameters Θ^0

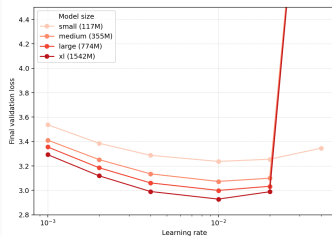
```

1:  $\mathbf{M}^0 \leftarrow \mathbf{0}$  ▷  $\mathbf{M}^t$  has the same block structure as  $\Theta^t$ 
2: for  $t = 1, 2, \dots$  do
3:   Sample stochastic gradient:  $\mathbf{G} \leftarrow \nabla F(\Theta^{t-1}; \xi)$ ,  $\xi \sim \mathbb{P}$ 
4:   Exponential moving average:  $\mathbf{M}^t \leftarrow \beta_1 \mathbf{M}^{t-1} + (1 - \beta_1) \mathbf{G}$ 
5:   Lookahead (Nesterov-type) momentum:  $\tilde{\mathbf{M}}^t \leftarrow \beta_2 \mathbf{M}^{t-1} + (1 - \beta_2) \mathbf{G}$ 
6:   if Descent under  $(1, \text{mean}) \rightarrow (q, \text{mean})$  then ▷  $q \geq 2$ 
7:     for  $i = 1$  to  $\ell$  do
8:        $\mathbf{W}_i^t \leftarrow \mathbf{W}_i^{t-1} - \eta^t \cdot \frac{(d_{\text{out}}^i)^{1/q}}{d_{\text{in}}^i} \cdot \text{colnorm}_q \left( \tilde{\mathbf{M}}_{\mathbf{W}_i^{t-1}}^t \right)$ 
9:        $\mathbf{b}_i^t \leftarrow \mathbf{b}_i^{t-1} - \eta^t \text{sign} \left( \tilde{\mathbf{M}}_{\mathbf{b}_i}^t \right)$  ▷ or steepest descent under  $(q, \text{mean})$  norm
10:    end for
11:   else if Descent under  $(p, \text{mean}) \rightarrow \infty$  then
12:     for  $i = 1$  to  $\ell$  do
13:        $\mathbf{W}_i^t \leftarrow \mathbf{W}_i^{t-1} - \eta^t \cdot \frac{1}{(d_{\text{in}}^i)^{1/p}} \cdot \text{rownorm}_p \left( \tilde{\mathbf{M}}_{\mathbf{W}_i^{t-1}}^t \right)$ 
14:        $\mathbf{b}_i^t \leftarrow \mathbf{b}_i^{t-1} - \eta^t \text{sign} \left( \tilde{\mathbf{M}}_{\mathbf{b}_i}^t \right)$ 
15:     end for
16:   end if
17: end for

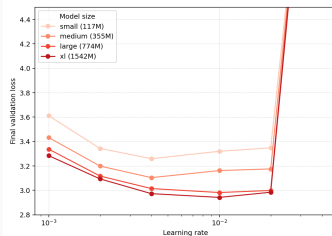
```

Learning Rate Transfer Across Model Scales

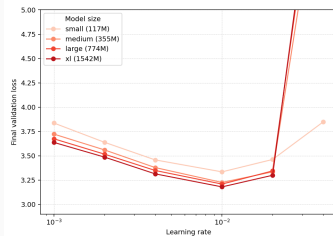
Architecture: GPT-2 family (Small $\rightarrow$ XL)



(a) MOGA ($p=1.5$)



(b) MOGA ($p=2$)



(c) MOGA ($p=3$)

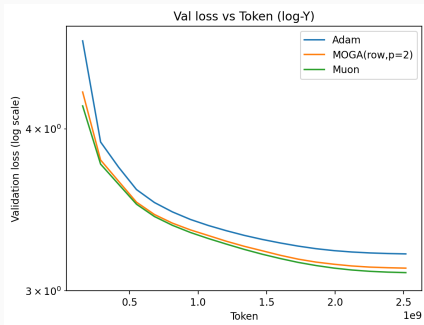
Figure 1: Learning Rate Transfer. Performance of MOGA from GPT-2 Small to GPT-XL. The optimal learning rate of MOGA is invariant to width.

LLM Pretraining — Standard Token Budget

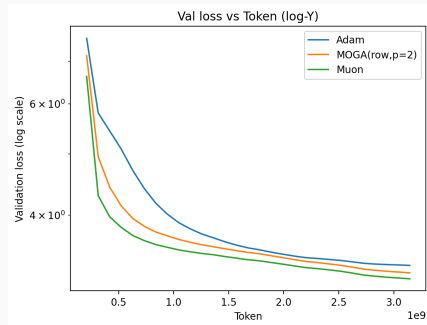
Model 1: LLaMA-130m (130M params) **Data 1:** C4 (Raffel et al., 2020).

Model 2: GPT-2 Small (117M params) **Data 2:** OpenWebText (Gokaslan et al., 2019).

Training Tokens: ~ 1 Chinchilla ($20\times$ params).



Llama-130m model

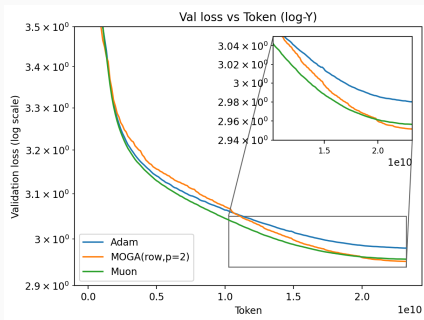


GPT-2 Small model

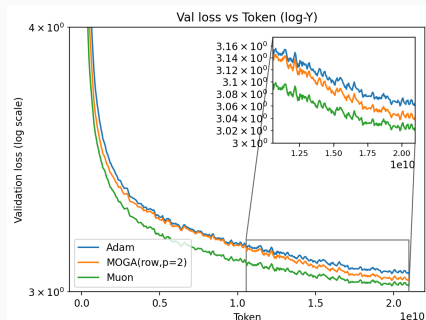
MOGA achieves almost the same speed as Muon in 1 chinchilla training.

LLM Pretraining – Large Token Budget

Training Tokens: ~ 8 Chinchilla ($160\times$ params).



Llama-130m model



GPT-2 Small model

MOGA achieves faster convergence in the later stage of training when the loss is small!

¹ We use Moonlight as an alternative version of Muon, which demonstrates great performance in a lot of benchmarks [Semenov et al, 2025; Wen et al, 2025].

- **Hyperparameter transfer** is necessary for LLM optimizer design.
- Steepest descent under different matrix operator norms (**Matrix Thinking**) offers a unified framework to understand successful existing optimizers (e.g., SignSGD, AdamW, Muon).
- **Selecting** the appropriate matrix operator norm is the key to designing new optimizers.
- Taking **width-independent Lipschitz/smoothness properties** as opt design principle.
- We introduce the mean-normalized $(p, \text{mean}) \rightarrow (q, \text{mean})$ operator norm to make the matrix operator norm play nicely together.
- This norm yields width-independent bounds when $q \geq 2p$, and naturally lead to width-aware learning rate rescheduling rules and new optimizers MOGA.
- When $p = 1, q = \infty$ we recover $\mu\mathbf{P}$ **scaling** in the Adam Optimizer.

- **Hyperparameter transfer** is necessary for LLM optimizer design.
- Steepest descent under different matrix operator norms (**Matrix Thinking**) offers a unified framework to understand successful existing optimizers (e.g., SignSGD, AdamW, Muon).
- **Selecting** the appropriate matrix operator norm is the key to designing new optimizers.
- Taking **width-independent Lipschitz/smoothness properties** as opt design principle.
- We introduce the mean-normalized $(p, \text{mean}) \rightarrow (q, \text{mean})$ operator norm to make the matrix operator norm play nicely together.
- This norm yields width-independent bounds when $q \geq 2p$, and naturally lead to width-aware learning rate rescheduling rules and new optimizers MOGA.
- When $p = 1, q = \infty$ we recover $\mu\mathbf{P}$ **scaling** in the Adam Optimizer.

- **Hyperparameter transfer** is necessary for LLM optimizer design.
- Steepest descent under different matrix operator norms (**Matrix Thinking**) offers a unified framework to understand successful existing optimizers (e.g., SignSGD, AdamW, Muon).
- **Selecting** the appropriate matrix operator norm is the key to designing new optimizers.
- Taking **width-independent Lipschitz/smoothness properties** as opt design principle.
- We introduce the mean-normalized $(p, \text{mean}) \rightarrow (q, \text{mean})$ operator norm to make the matrix operator norm play nicely together.
- This norm yields width-independent bounds when $q \geq 2p$, and naturally lead to width-aware learning rate rescheduling rules and new optimizers MOGA.
- When $p = 1, q = \infty$ we recover $\mu\mathbf{P}$ **scaling** in the Adam Optimizer.

- **Hyperparameter transfer** is necessary for LLM optimizer design.
- Steepest descent under different matrix operator norms (**Matrix Thinking**) offers a unified framework to understand successful existing optimizers (e.g., SignSGD, AdamW, Muon).
- **Selecting** the appropriate matrix operator norm is the key to designing new optimizers.
- Taking **width-independent Lipschitz/smoothness properties** as opt design principle.
- We introduce the mean-normalized $(p, \text{mean}) \rightarrow (q, \text{mean})$ operator norm to make the matrix operator norm play nicely together.
- This norm yields width-independent bounds when $q \geq 2p$, and naturally lead to width-aware learning rate rescheduling rules and new optimizers MOGA.
- When $p = 1, q = \infty$ we recover μP **scaling** in the Adam Optimizer.

- **Hyperparameter transfer** is necessary for LLM optimizer design.
- Steepest descent under different matrix operator norms (**Matrix Thinking**) offers a unified framework to understand successful existing optimizers (e.g., SignSGD, AdamW, Muon).
- **Selecting** the appropriate matrix operator norm is the key to designing new optimizers.
- Taking **width-independent Lipschitz/smoothness properties** as opt design principle.
- We introduce the mean-normalized $(p, \text{mean}) \rightarrow (q, \text{mean})$ operator norm to make the matrix operator norm play nicely together.
- This norm yields width-independent bounds when $q \geq 2p$, and naturally lead to width-aware learning rate rescheduling rules and new optimizers MOGA.
- When $p = 1, q = \infty$ we recover μP scaling in the Adam Optimizer.

- **Hyperparameter transfer** is necessary for LLM optimizer design.
- Steepest descent under different matrix operator norms (**Matrix Thinking**) offers a unified framework to understand successful existing optimizers (e.g., SignSGD, AdamW, Muon).
- **Selecting** the appropriate matrix operator norm is the key to designing new optimizers.
- Taking **width-independent Lipschitz/smoothness properties** as opt design principle.
- We introduce the mean-normalized $(p, \text{mean}) \rightarrow (q, \text{mean})$ operator norm to make the matrix operator norm play nicely together.
- This norm yields width-independent bounds when $q \geq 2p$, and naturally lead to width-aware learning rate rescheduling rules and new optimizers MOGA.
- When $p = 1, q = \infty$ we recover μP scaling in the Adam Optimizer.

- **Hyperparameter transfer** is necessary for LLM optimizer design.
- Steepest descent under different matrix operator norms (**Matrix Thinking**) offers a unified framework to understand successful existing optimizers (e.g., SignSGD, AdamW, Muon).
- **Selecting** the appropriate matrix operator norm is the key to designing new optimizers.
- Taking **width-independent Lipschitz/smoothness properties** as opt design principle.
- We introduce the mean-normalized $(p, \text{mean}) \rightarrow (q, \text{mean})$ operator norm to make the matrix operator norm play nicely together.
- This norm yields width-independent bounds when $q \geq 2p$, and naturally lead to width-aware learning rate rescheduling rules and new optimizers MOGA.
- When $p = 1, q = \infty$ we recover $\mu\mathbf{P}$ **scaling** in the Adam Optimizer.

Thank you for your listening!
Any questions?

- The change in feature

$$\|y_I\|_2 = \Theta(\sqrt{\text{fanout}}), \text{ and } \|\Delta y_I\|_2 = \Theta(\sqrt{\text{fanout}})$$

- As a result for parameter

$$\|W_I\|_2 = \Theta\left(\sqrt{\frac{\text{fanout}}{\text{fanin}}}\right), \text{ and } \|\Delta W_I\|_2 = \Theta\left(\sqrt{\frac{\text{fanout}}{\text{fanin}}}\right)$$

Yang G, Simon J B, Bernstein J. A spectral condition for feature learning. arXiv preprint arXiv:2310.17813, 2023.