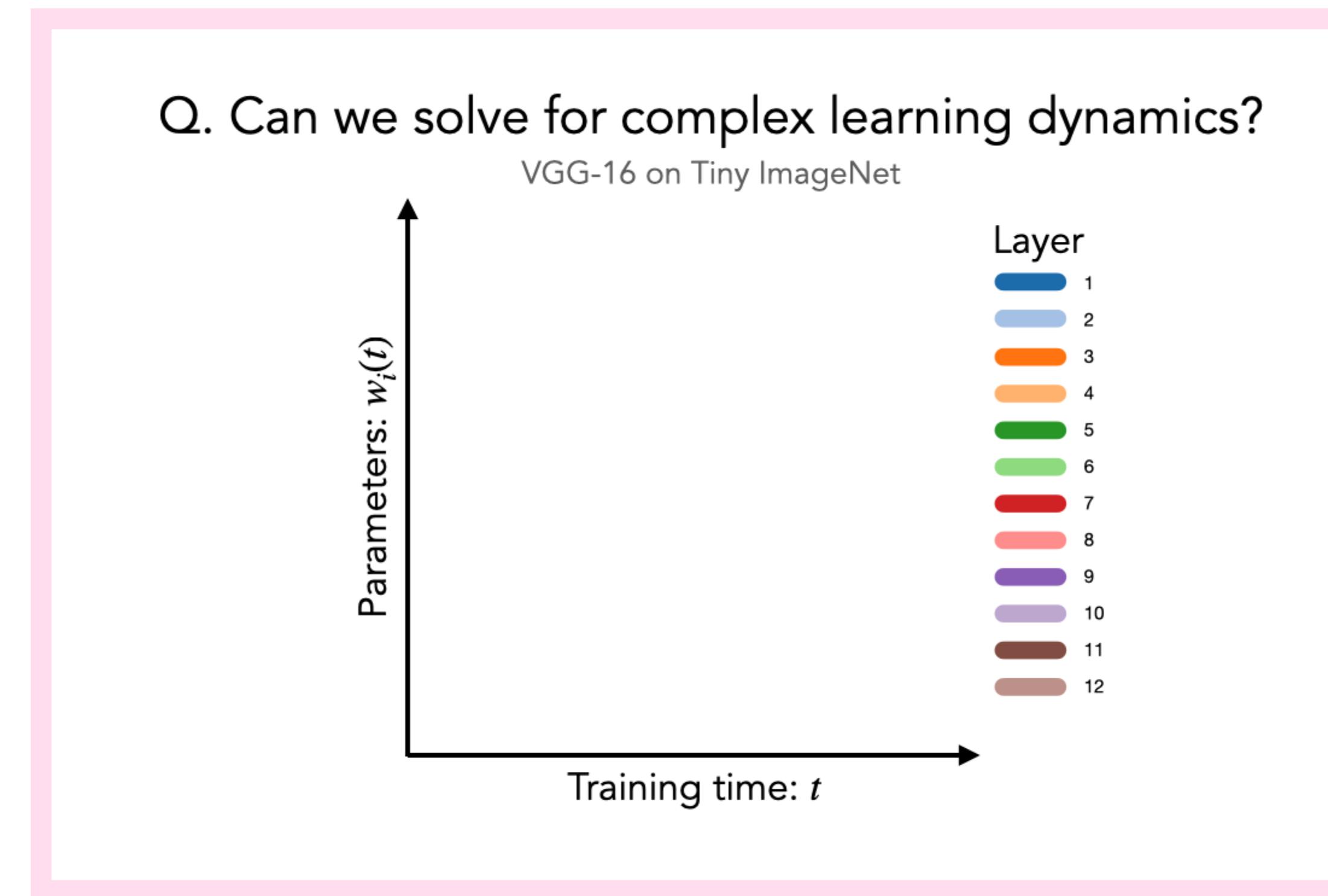


# Neural Mechanics: Symmetry and Broken Conservation Laws in Deep Learning Dynamics



Daniel Kunin\*, Javier Sagastuy-Brena, Surya Ganguli, Daniel L.K. Yamins, Hidenori Tanaka\*  
(\* equal contribution)

Stanford University, NTT Physics & Informatics Lab

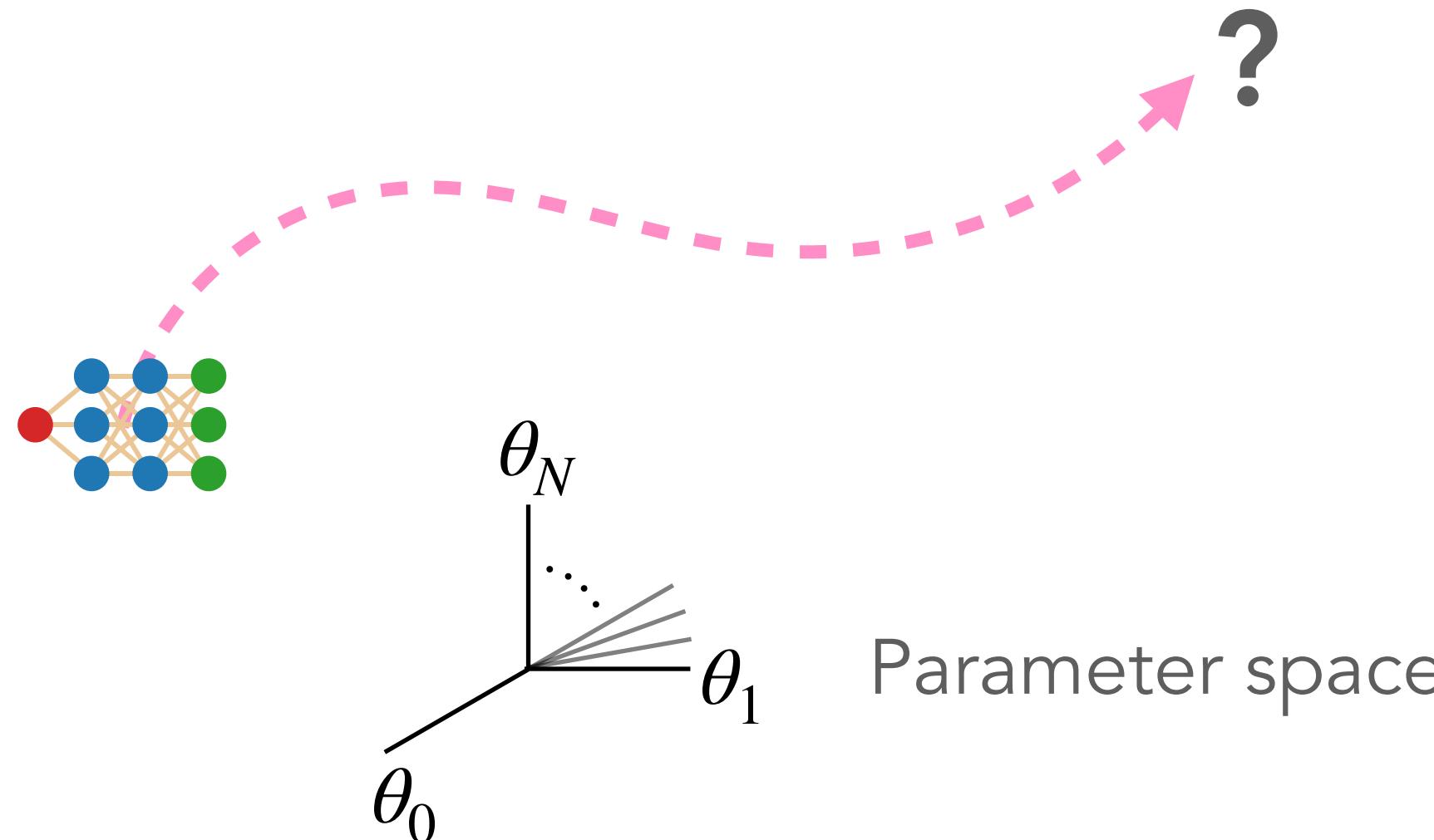


Deep learning has been successful, but its inner-workings are still mysterious

Myriads of design choices of the deep learning system  
shapes the trajectory that the millions of parameters take during training

### Architecture

- ReLU or tanh?
- Batch Normalization?
- SoftMax?
- Convolution?
- Residual connection?



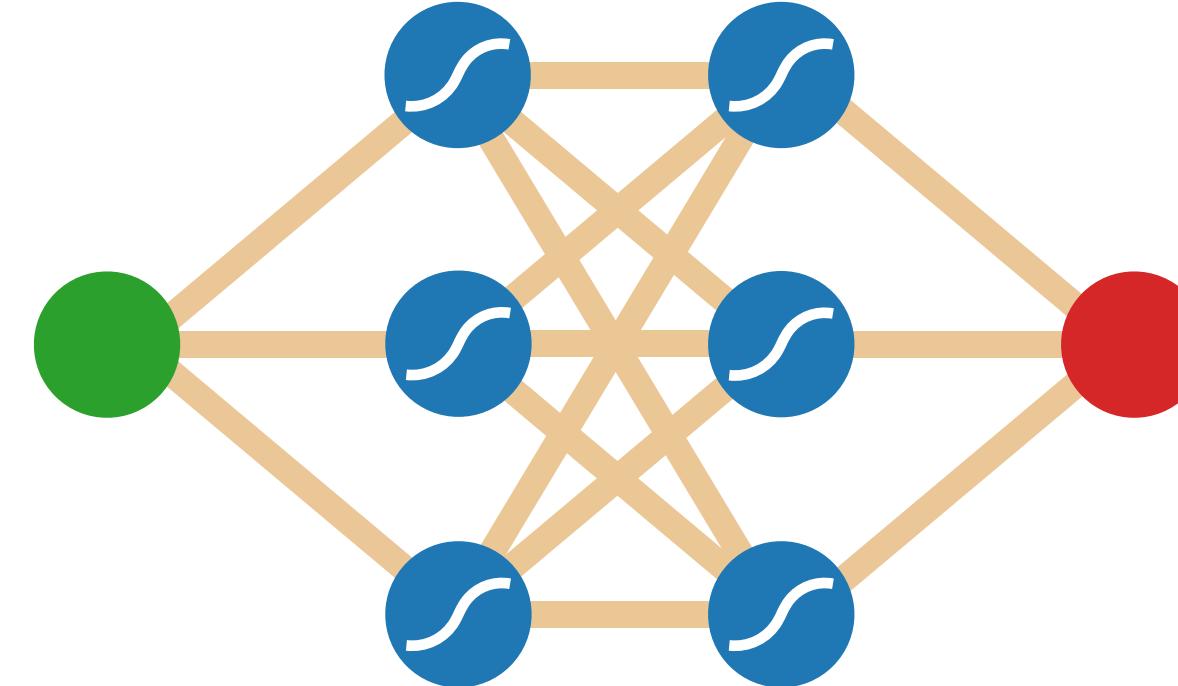
### Optimizer

- How much weight decay?
- How much momentum?
- Learning rate schedule?
- Batch-size?
- Adaptive gradient?

Researchers and practitioners largely depend on try & error based heuristic search.  
Better understanding of the mechanism is foundational in principled exploration of the vast design space.

Q. What, if anything, can we quantitatively understand about the learning dynamics of state-of-the-art deep learning models driven by real-world datasets?

Q. What, if anything, can we quantitatively understand about the learning dynamics of state-of-the-art deep learning models driven by real-world datasets?



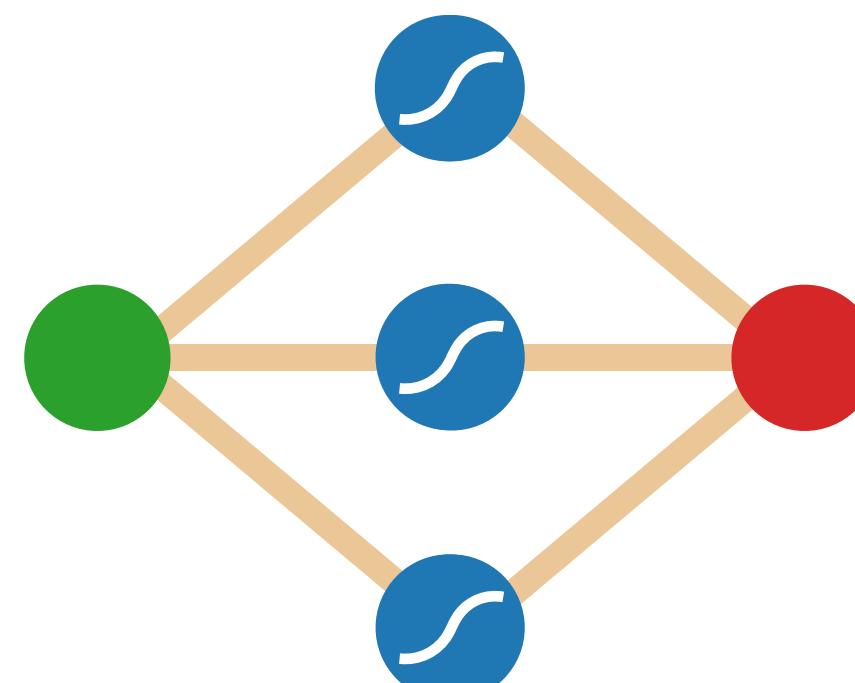
This question is difficult because of...

1. millions of parameters
2. compositional non-linear functions
3. discrete updates by random batches of data

Existing works have simplified the problem by making major assumptions on the architecture...

Single Hidden Layer

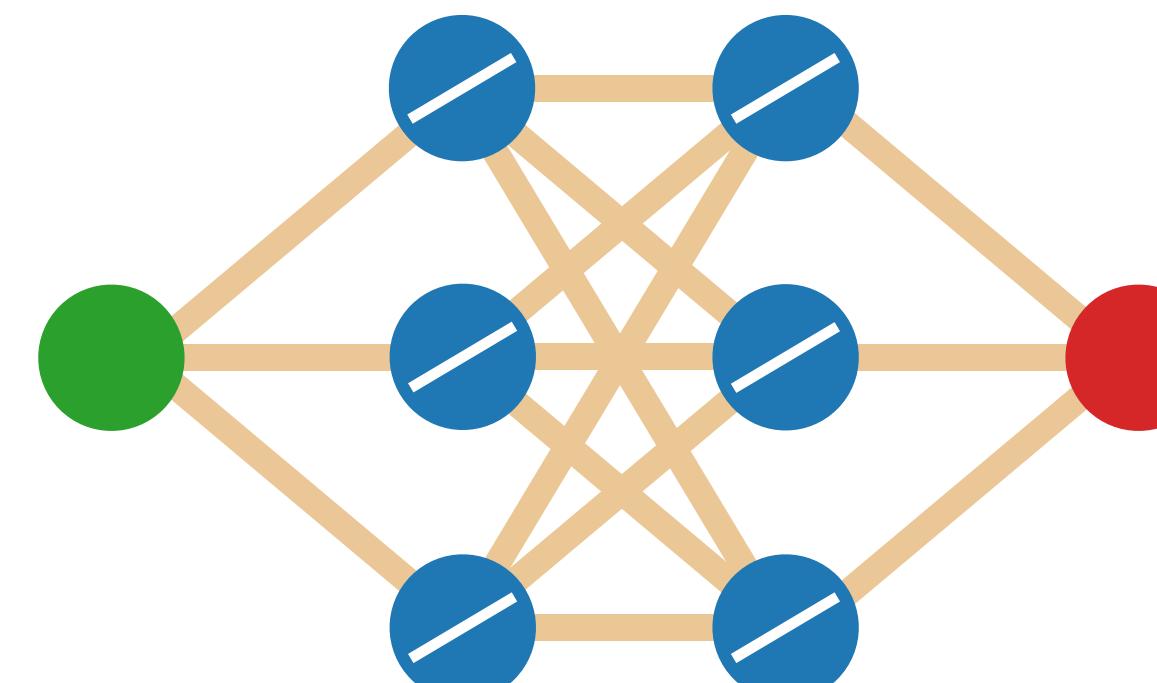
$$y = \theta^{[2]} f(\theta^{[1]} x)$$



David Saad and Sara Solla. Dynamics of on-line gradient descent learning for multilayer neural networks. 1995.

Linear Networks

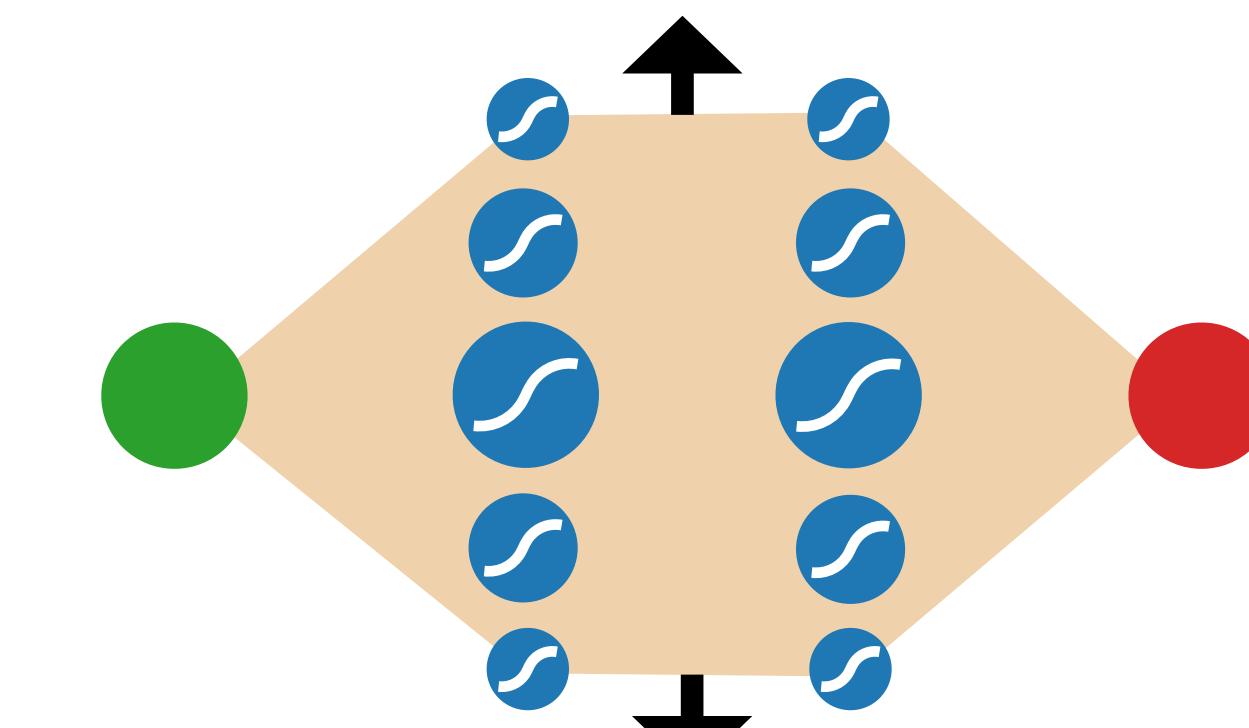
$$y = \theta^{[L]} \dots \theta^{[2]} \theta^{[1]} x$$



Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. 2013.

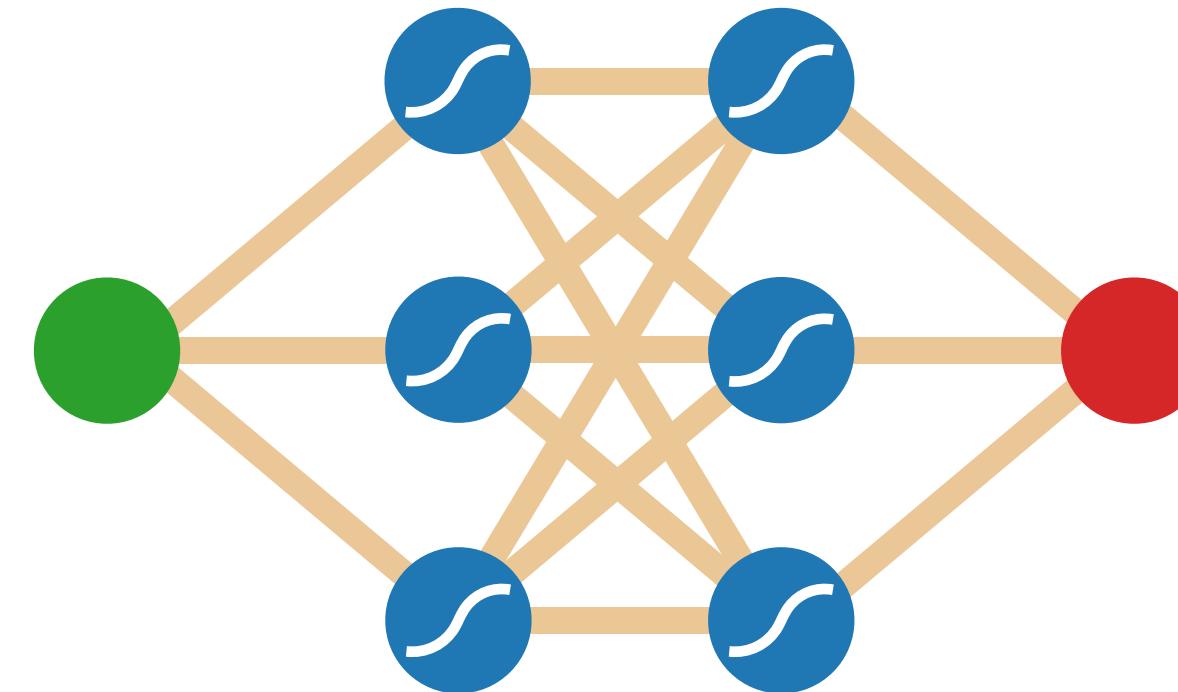
Infinitely Wide

$$\theta^{[l]} \in R^{N^{[l]} \times N^{[l]}}, N^{[l]} \rightarrow \infty$$



Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. 2018.

Q. What, if anything, can we quantitatively understand about the learning dynamics of state-of-the-art deep learning models driven by real-world datasets?

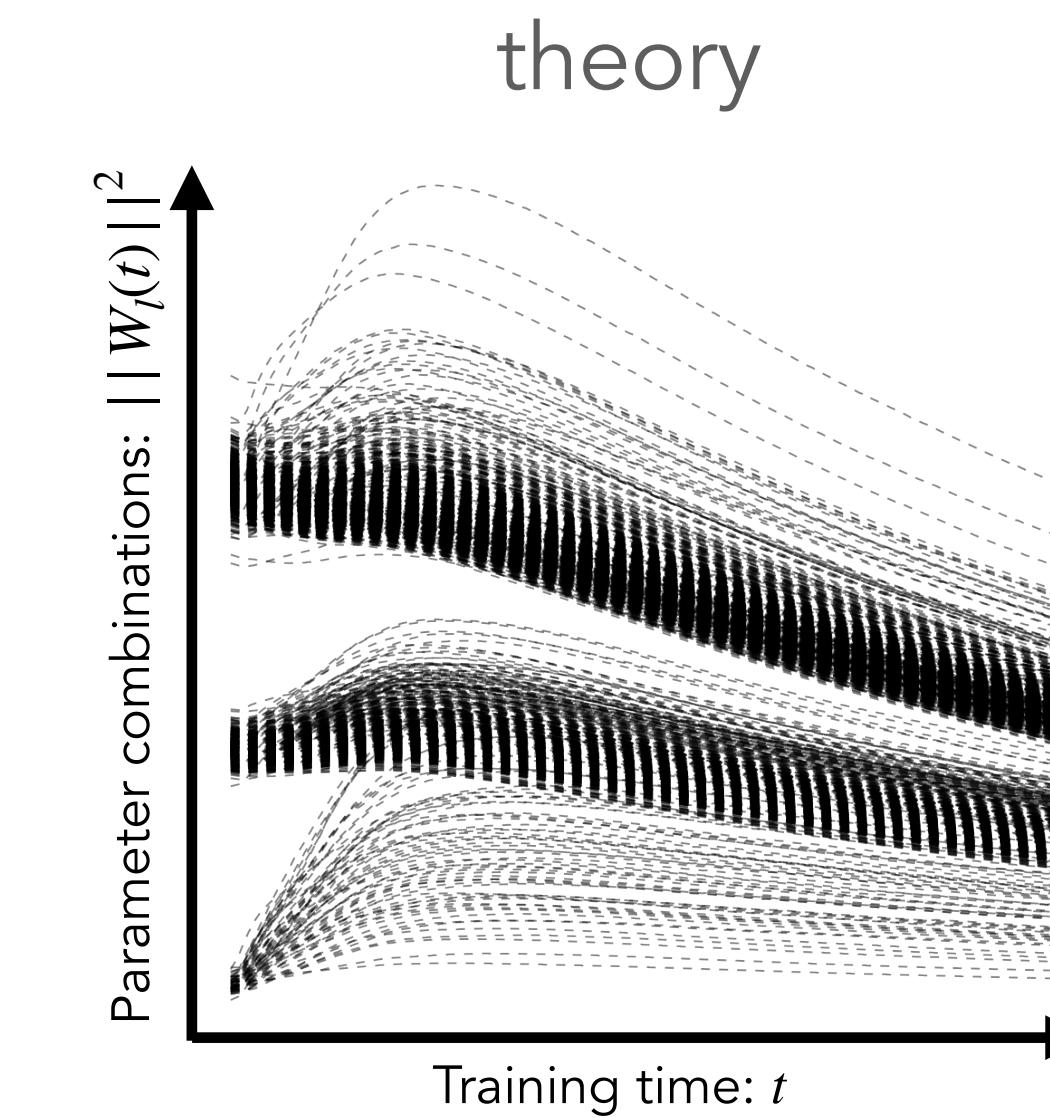
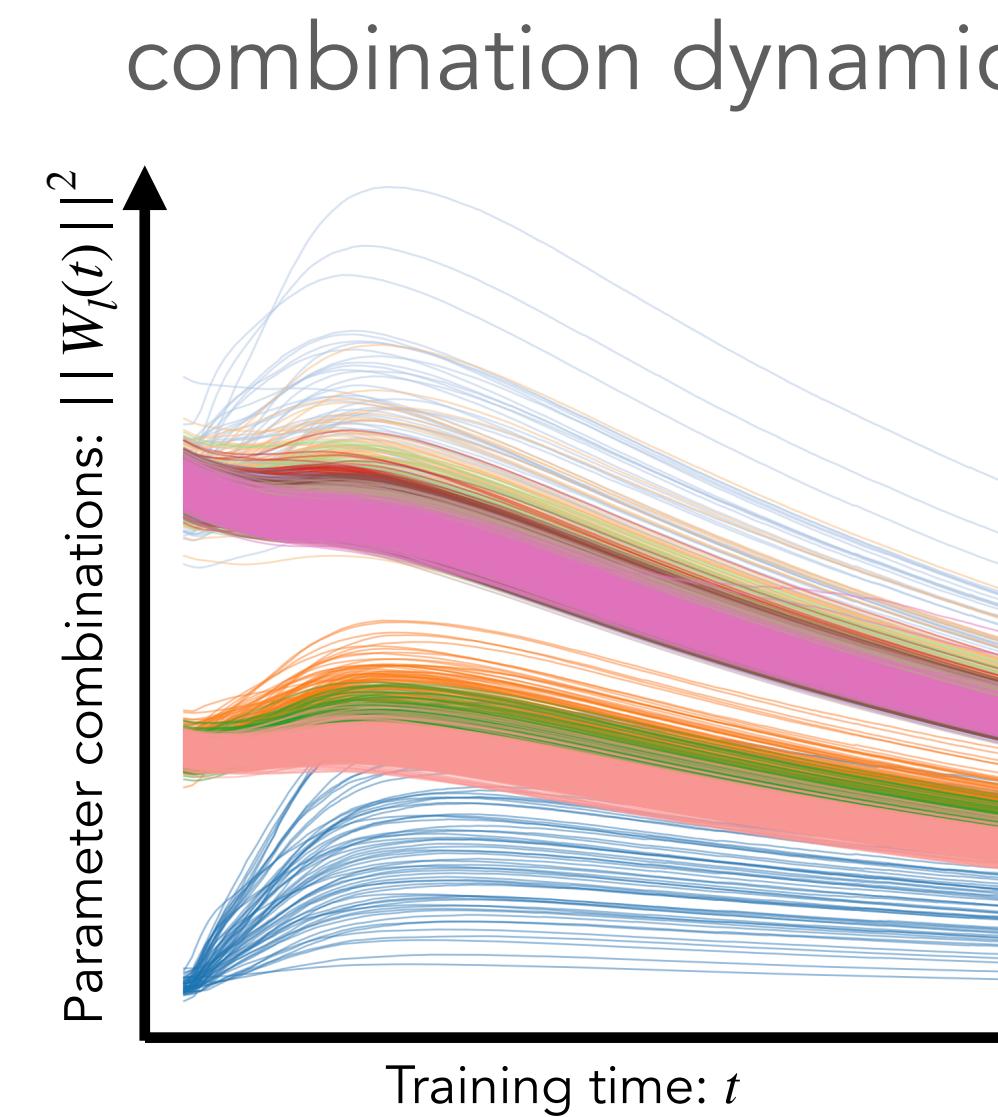
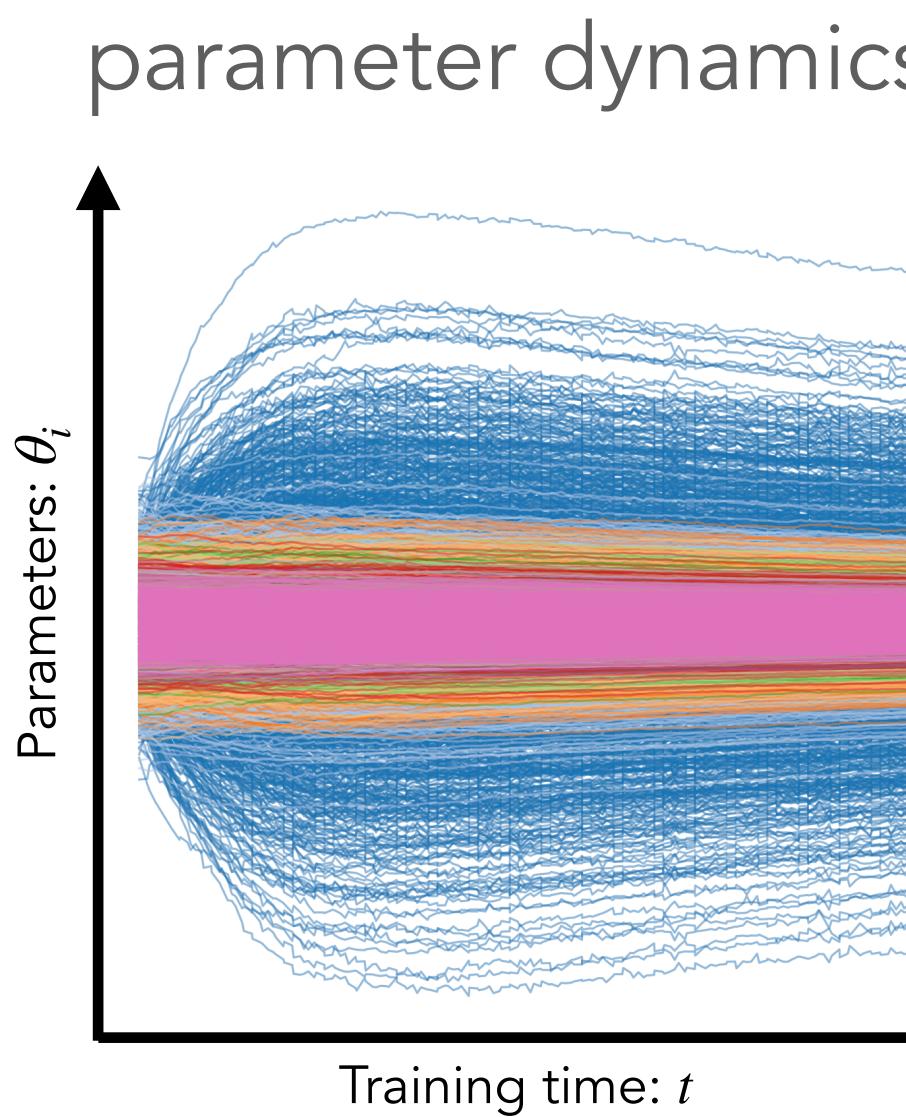


In this work we **don't introduce simplifying assumptions** on the architecture or optimizer!

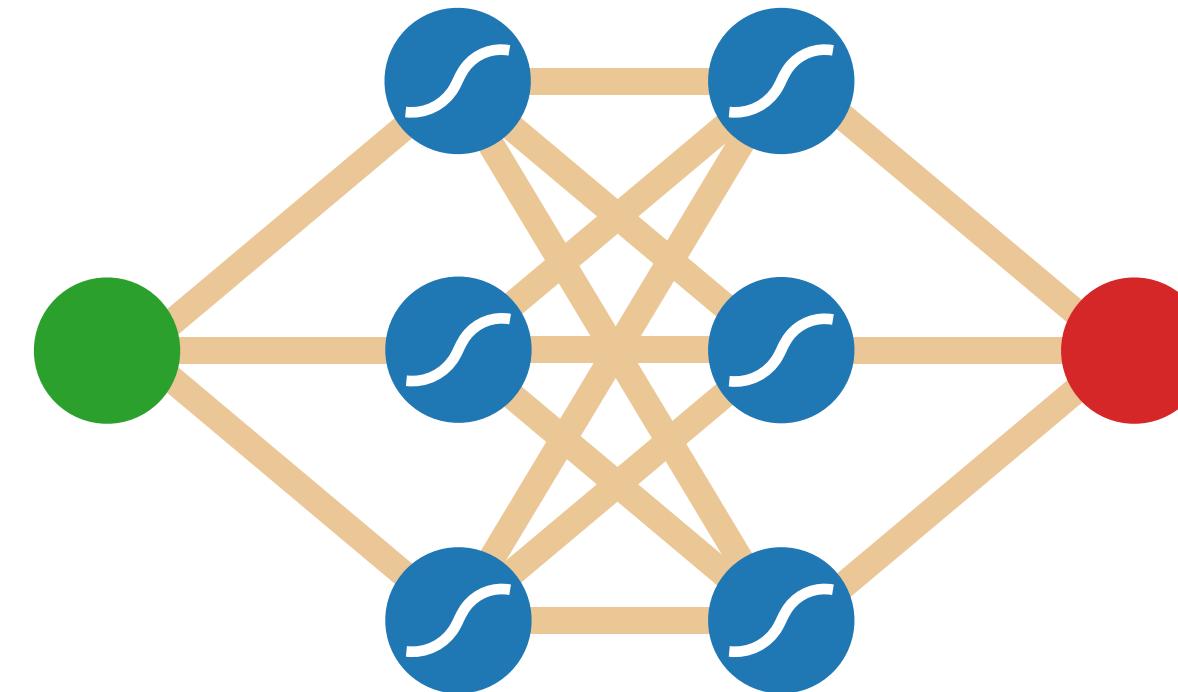
Rather we identify and solve the simpler dynamics of **parameter combinations**.

VGG16

- conv. 1
- conv. 2
- conv. 3
- conv. 4
- conv. 5
- conv. 6
- conv. 7
- conv. 8
- conv. 9
- conv. 10
- conv. 11
- conv. 12

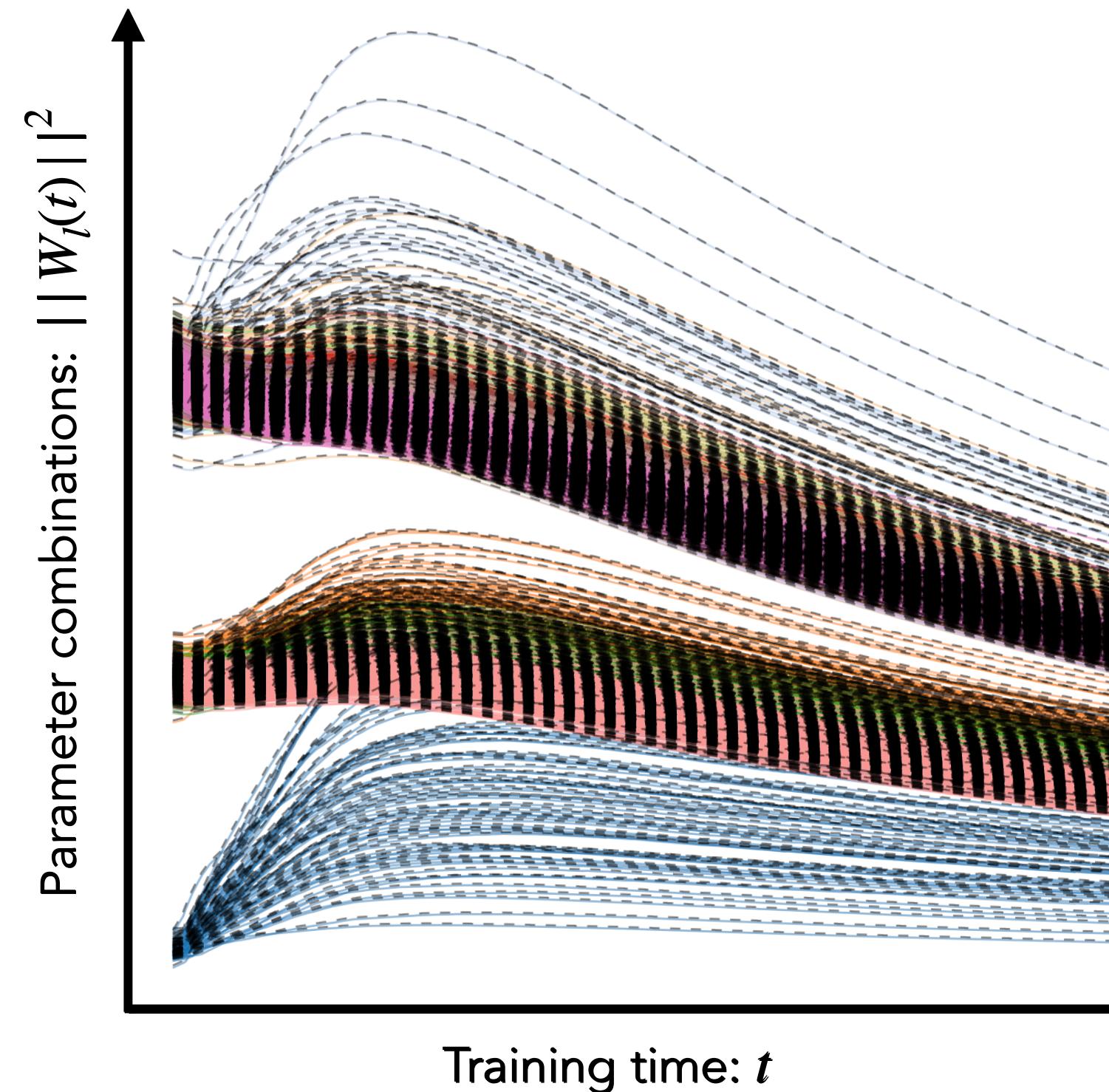


Q. What, if anything, can we quantitatively understand about the learning dynamics of state-of-the-art deep learning models driven by real-world datasets?



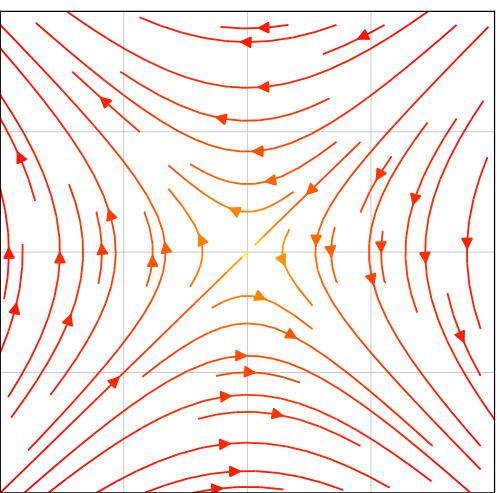
In this work we **don't introduce simplifying assumptions** on the architecture or optimizer!

Rather we identify and solve the simpler dynamics of **parameter combinations**.

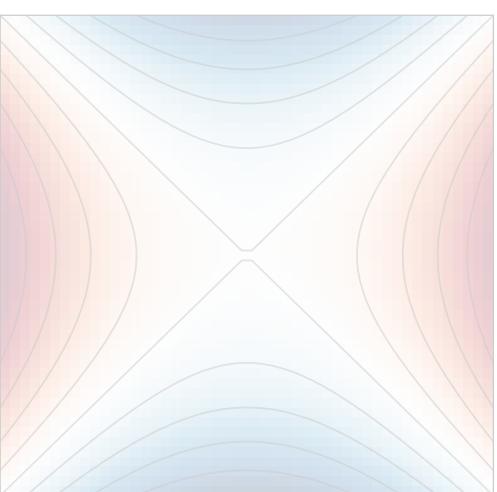


**Our theory matches  
experiment exactly!**

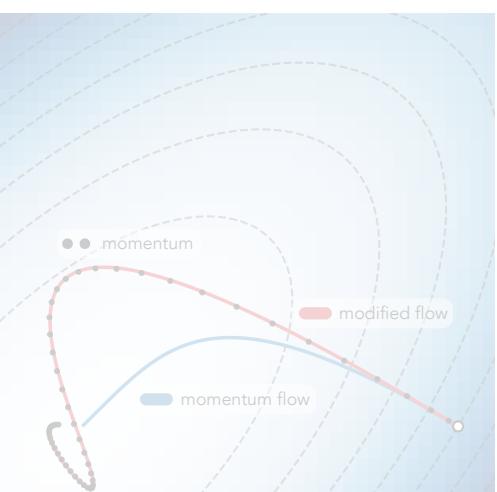
# Q. Can we solve for complex learning dynamics of real deep learning models?



Part 1. Symmetry in the Loss Constrain Gradient and Hessian Geometries



Part 2. Symmetry Leads to Conservation Laws Under Gradient Flow



Part 3. A Realistic Continuous Model for Stochastic Gradient Descent



Part 4. Combining Symmetry and Modified Flow to Derive Learning Dynamics

# Symmetry Constrain Gradient and Hessian Geometries

**Symmetry:** A function  $f(\theta)$  posses a symmetry if it invariant under the action  $\theta \mapsto \psi(\theta, \alpha)$  of a group  $G$  on the parameter vector  $\theta$ , i.e. if  $f(\psi(\theta, \alpha)) = f(\theta)$  for any  $(\theta, \alpha)$ .

**Geometric constraints:** If a function  $f(\theta)$  posses a differentiable symmetry, then

Gradient

$$\partial_\alpha f(\psi) = \langle \nabla f, \partial_\alpha \psi \rangle = 0$$

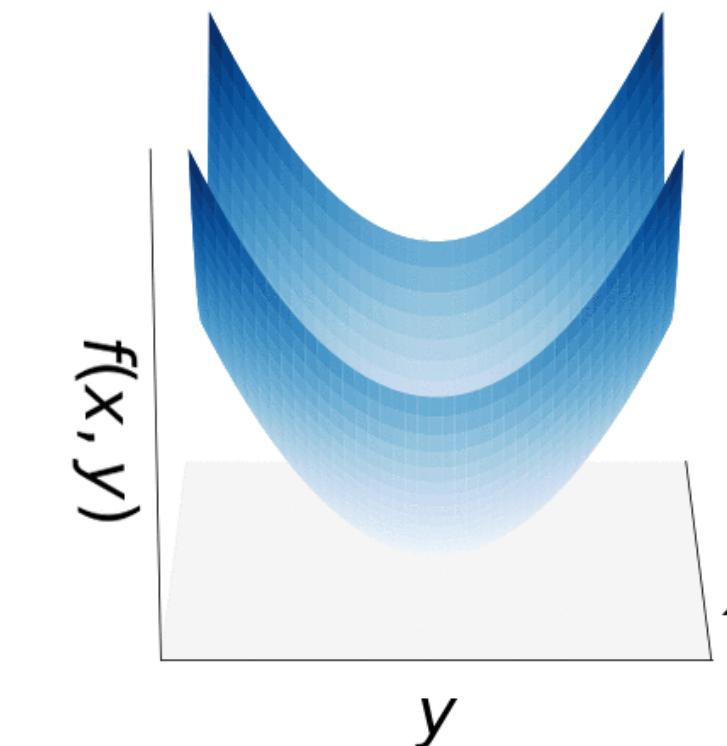
Hessian

$$\partial_\theta \partial_\alpha f(\psi) = \mathbf{H} f \partial_\theta \psi \partial_\alpha \psi + \partial_\theta \partial_\alpha \psi \nabla f = 0$$

**Example:**  $f(x, y) = x^2 + y^2$

Step 1. Identify symmetry:

Rotation:  $\phi(x, y, \alpha) = \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$

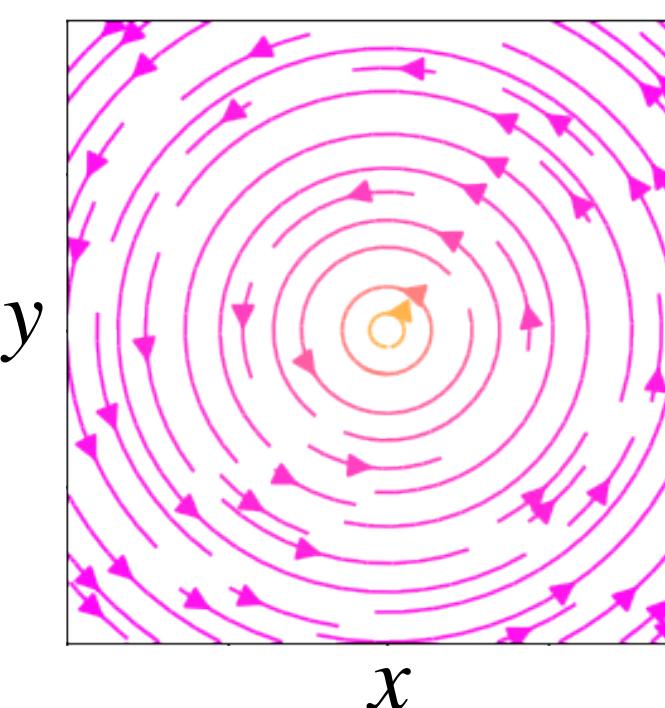
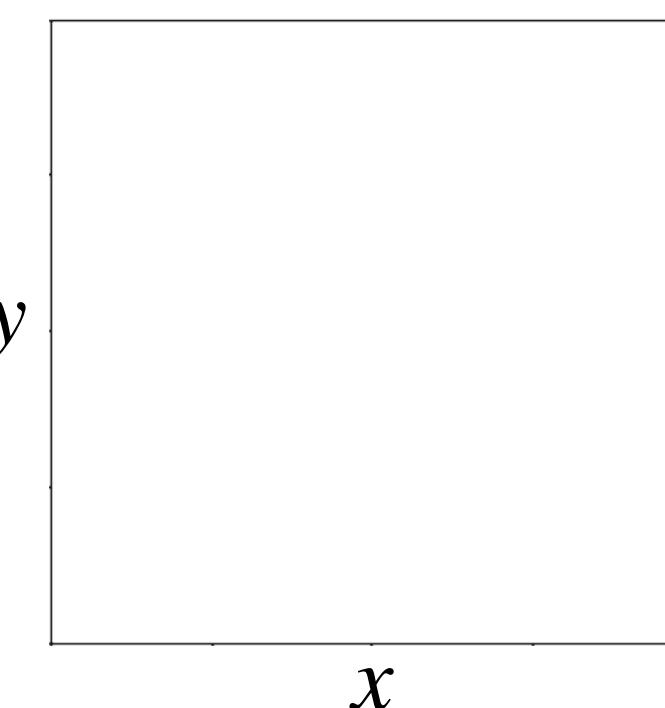
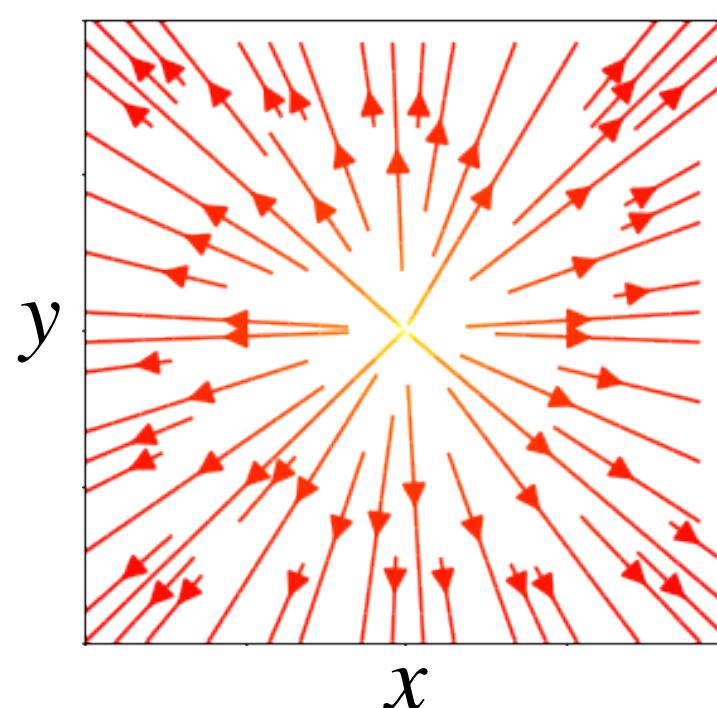


Step 2. Evaluate gradient at identity:

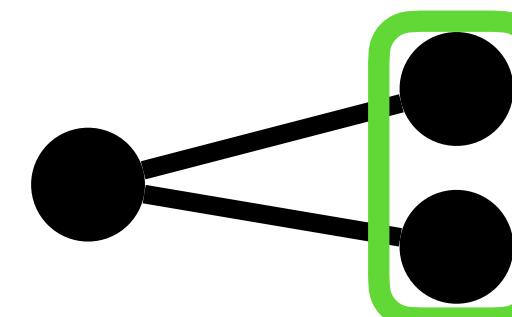
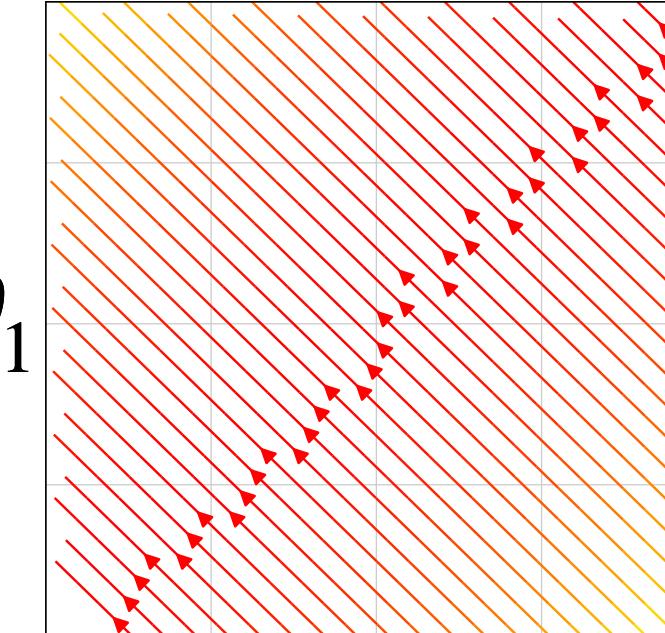
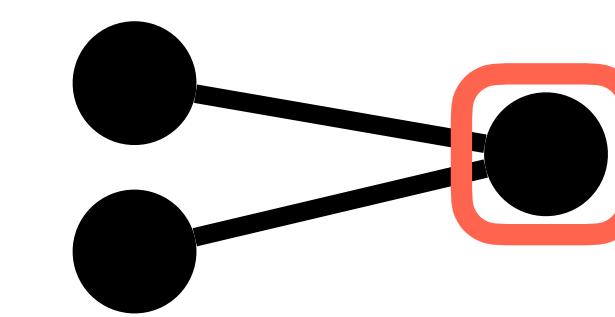
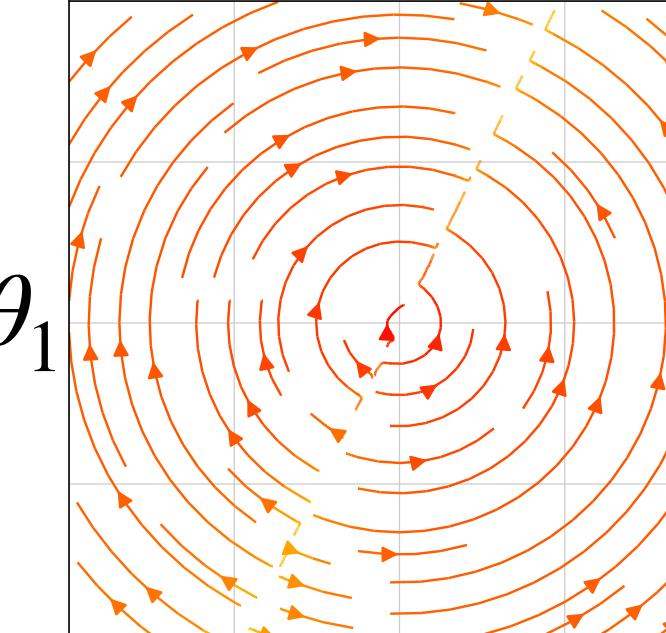
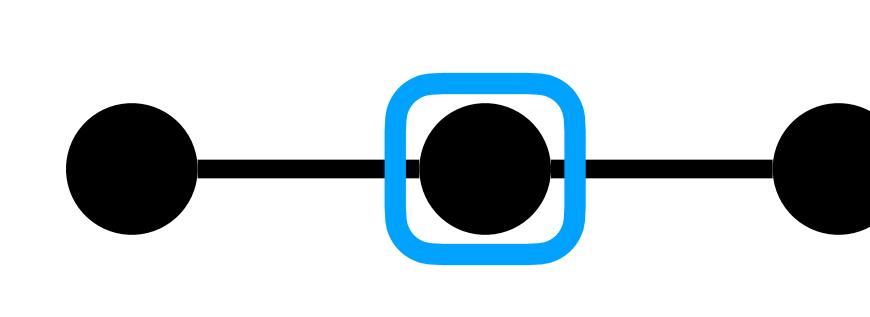
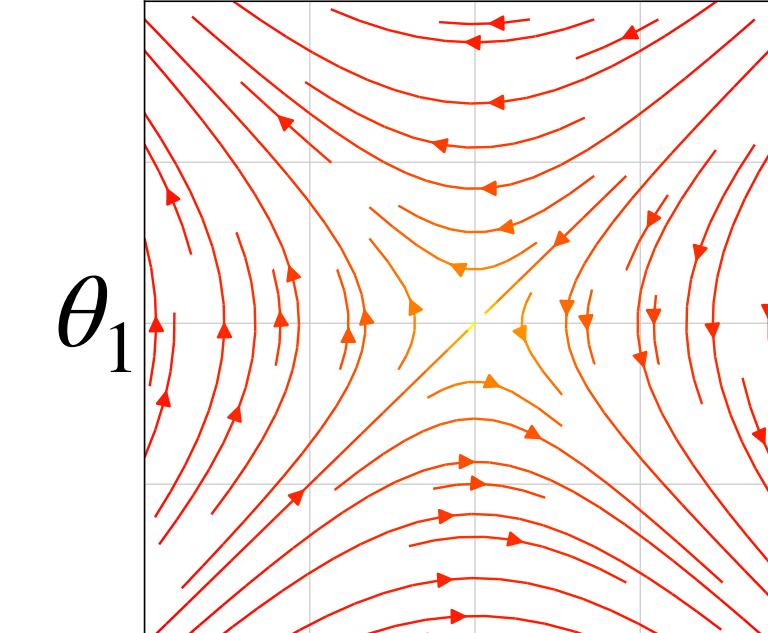
$$\nabla f = 2 \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\langle \nabla f, \partial_\alpha \phi \rangle = 0$$

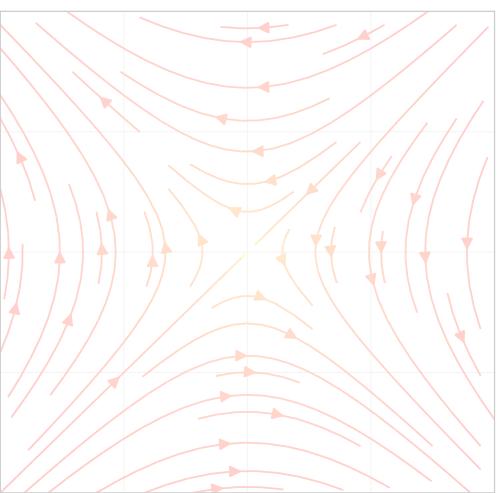
$$\partial_\alpha \phi|_{\alpha=0} = \begin{bmatrix} -y \\ x \end{bmatrix}$$



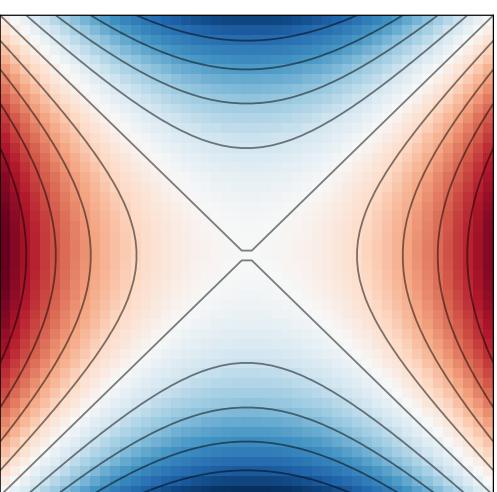
# Symmetry resides in all over the modern deep network architectures

	<b>Translation</b>	<b>Scale</b>	<b>Rescale</b>
<u>Symmetry</u>	$\theta \mapsto \theta + \alpha 1$ $\alpha \in \mathbb{R}$	$\theta \mapsto \alpha \theta$ $\alpha \in \mathbb{R}^+$	$(\theta_1, \theta_2) \mapsto (\alpha \theta_1, \alpha^{-1} \theta_2)$ $\alpha \in \mathbb{R}^+$
<u>Example</u>	softmax	batchnorm	ReLU
	$\sigma(\theta x)_i = \frac{e^{\theta_i x}}{\sum_j e^{\theta_j x}}$	$BN(\theta x) = \frac{\theta x - E[\theta x]}{\sqrt{\text{Var}(\theta x)}}$	$\theta_2 \text{ReLU}(\theta_1 x) = \theta_2 \max(0, \theta_1 x)$
<u>Gradient</u>	$\langle g, 1 \rangle = 0$	$\langle g, \theta \rangle = 0$	$\langle g_1, \theta_1 \rangle = \langle g_2, \theta_2 \rangle$
<u>Hessian</u>	$\langle H, 1 \rangle = 0$	$\langle H, \theta \rangle = -g$	$H(\theta_1 - \theta_2) + g_1 - g_2 = 0$
<u>Visualization</u>	 	 	 

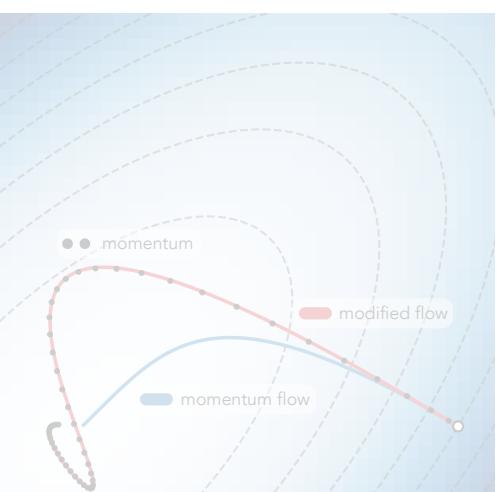
# Q. Can we solve for complex learning dynamics of real deep learning models?



Part 1. Symmetry in the Loss Constrain Gradient and Hessian Geometries



Part 2. Symmetry Leads to Conservation Laws Under Gradient Flow



Part 3. A Realistic Continuous Model for Stochastic Gradient Descent



Part 4. Combining Symmetry and Modified Flow to Derive Learning Dynamics

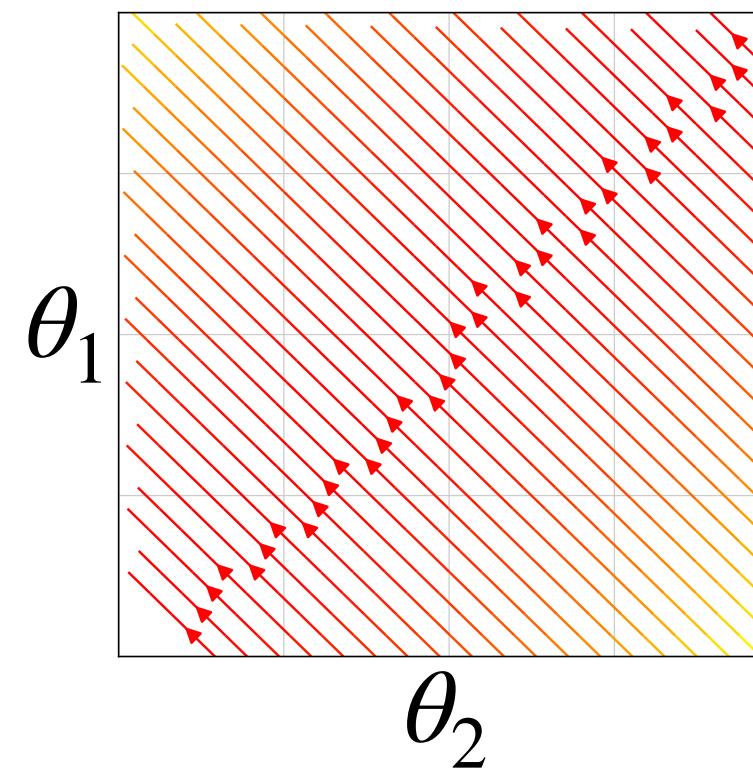
# Symmetry Leads to Conservation Laws Under Gradient Flow

**Gradient flow:** The gradient descent update  $\theta^{(n+1)} = \theta^{(n)} - \eta g(\theta^{(n)})$  with learning rate  $\eta$  is a forward Euler discretization of the ODE known as gradient flow:

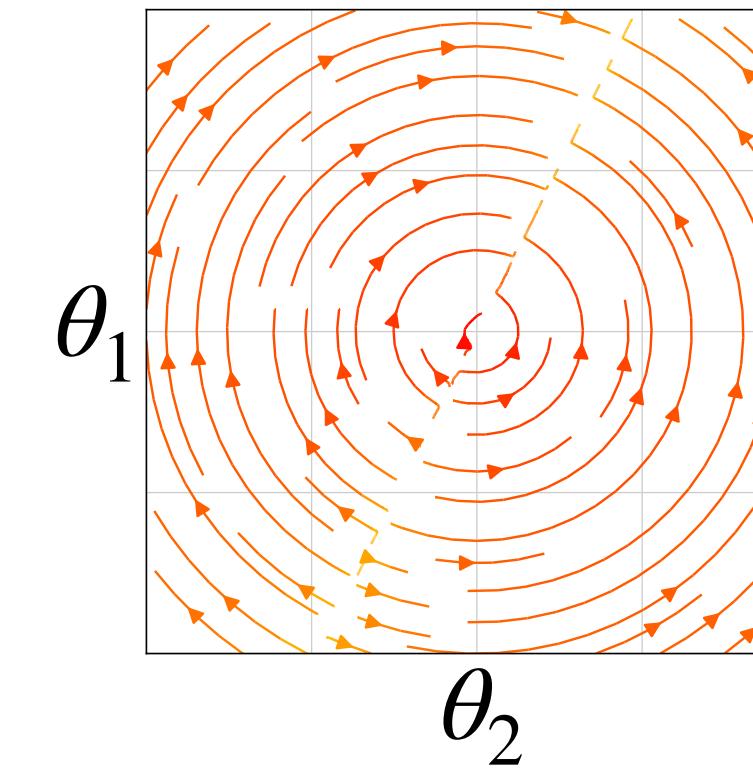
$$\frac{d\theta}{dt} = -g(\theta)$$

How do these learning dynamics interact with the geometric properties introduced by symmetry?

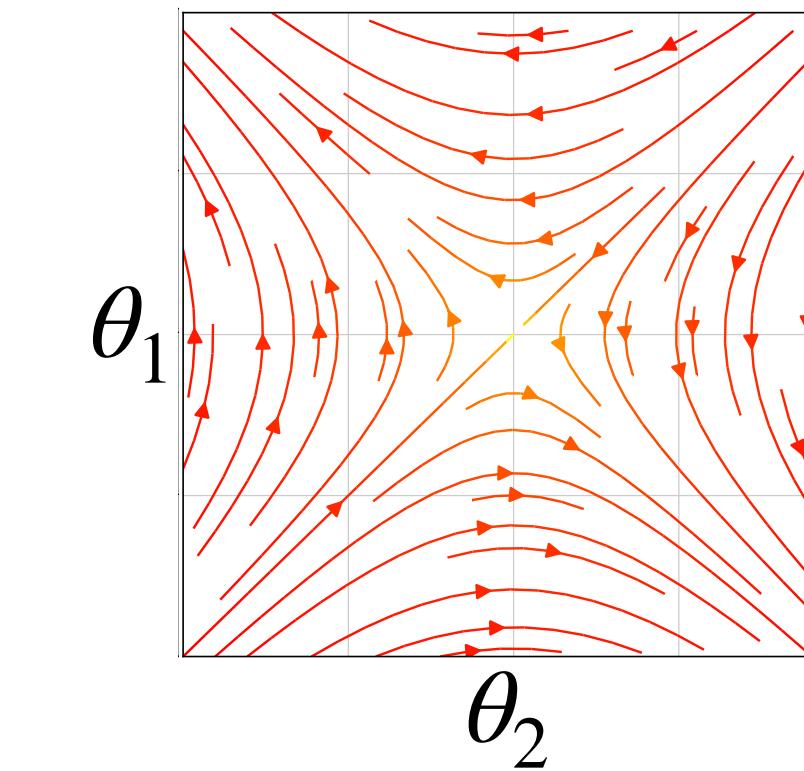
Translation



Scale



Rescale



$$\langle \theta_A(t), 1 \rangle = \langle \theta_A(0), 1 \rangle$$

$$|\theta_A(t)|^2 = |\theta_A(0)|^2$$

$$|\theta_{A_1}(t)|^2 - |\theta_{A_2}(t)|^2 = |\theta_{A_1}(0)|^2 - |\theta_{A_2}(0)|^2$$



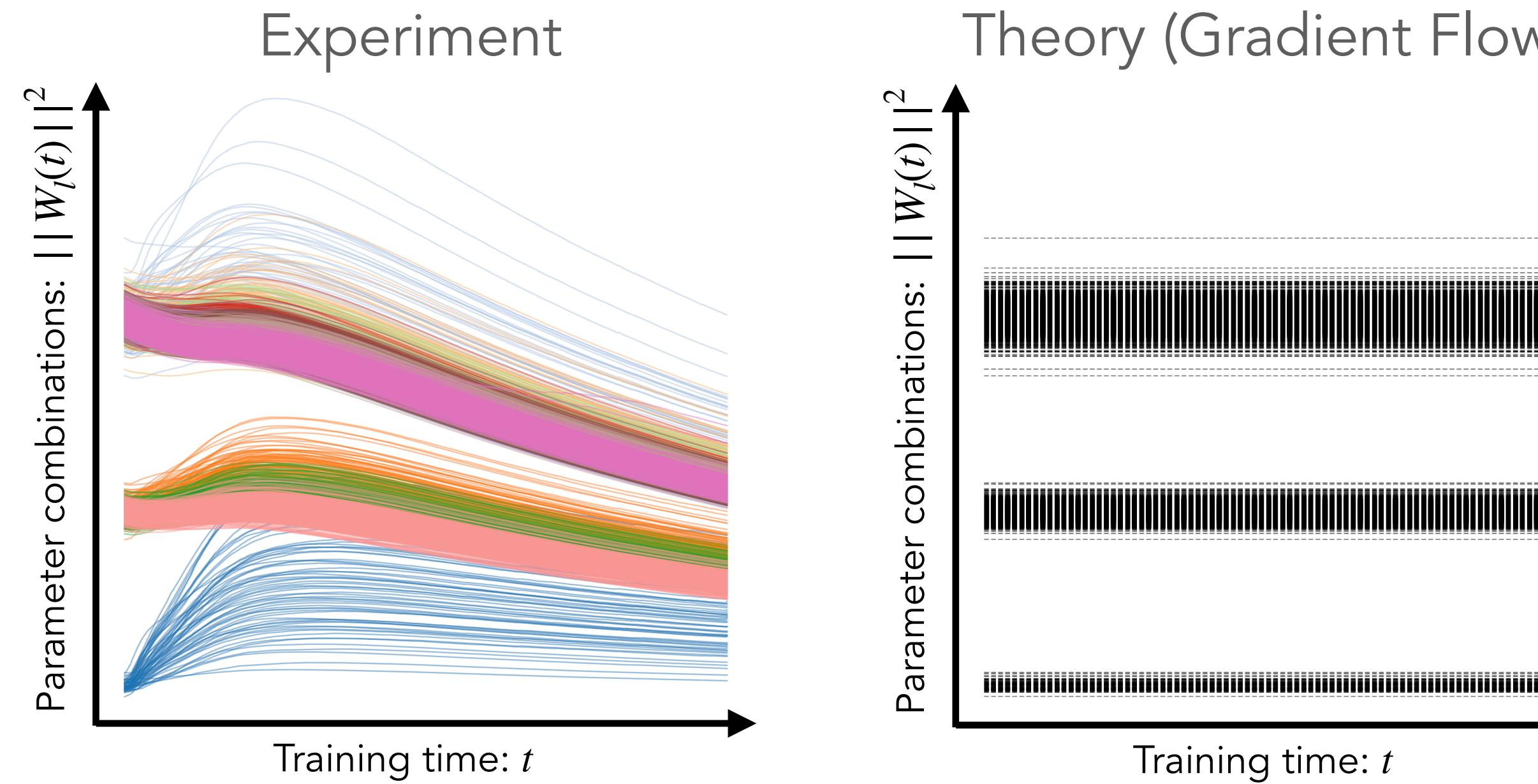
Emmy Noether (1882 - 1935)

**A version of Noether's Theorem:** Every symmetry\* of a network architecture has a corresponding conserved quantity through training under gradient flow. Projecting the gradient flow dynamics onto the generator vector field generates an ODE, whose solution is a conservation law.

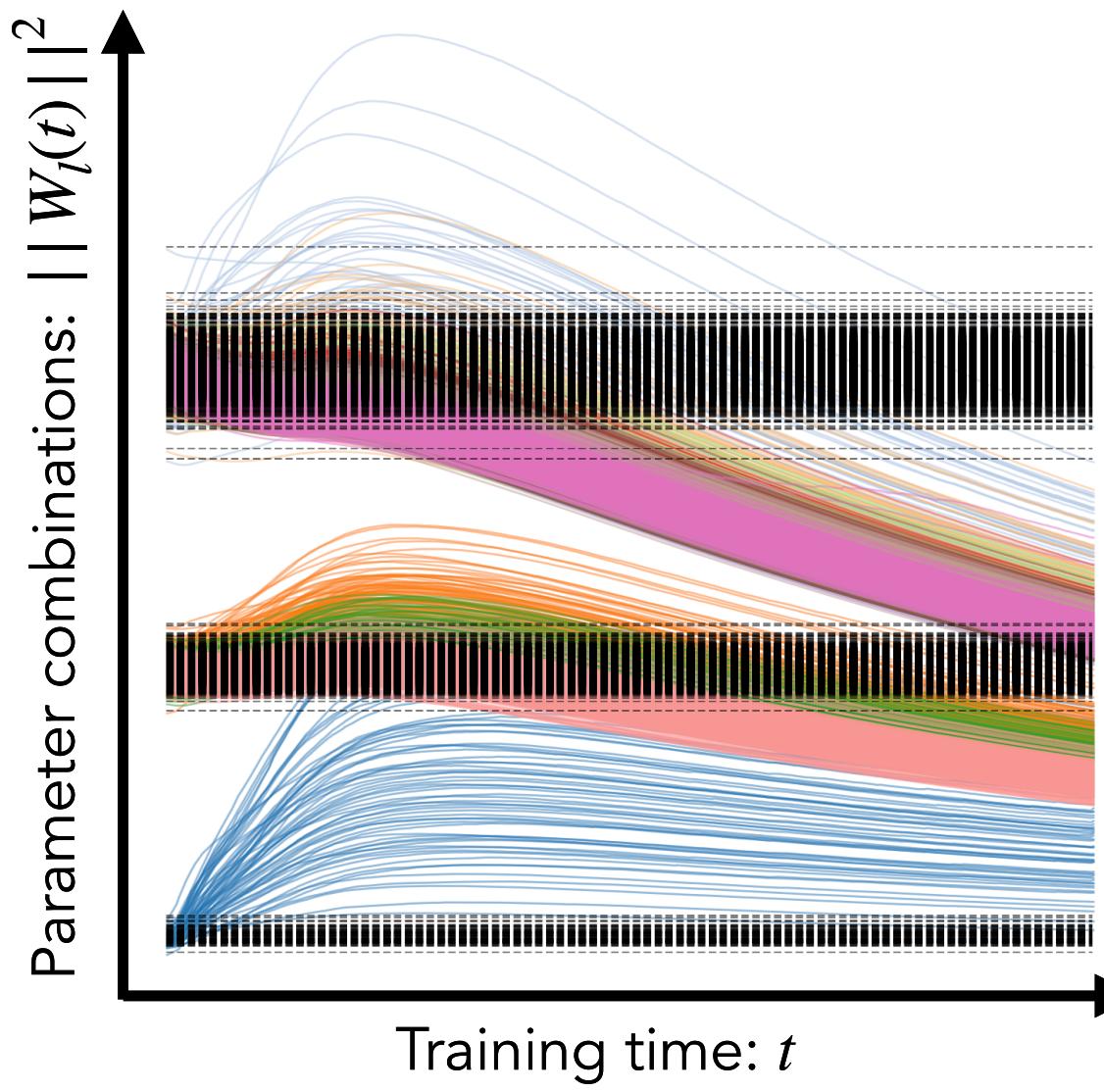
$$\frac{d}{dt} \left\langle \theta, \partial_\alpha \psi \right\rangle = 0$$

\*satisfying a mild assumption

# Does this theory agree with empirics?



# No, conservation laws are broken empirically!

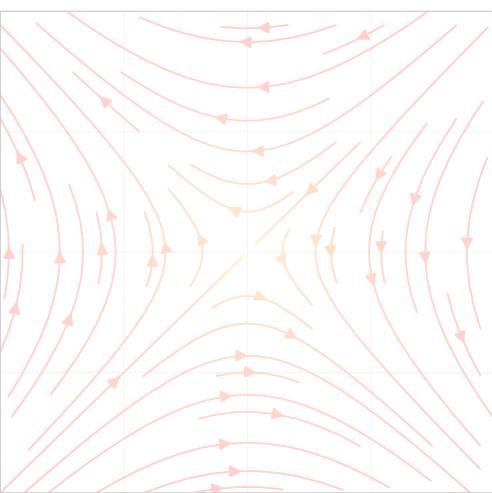


## Q. Why?

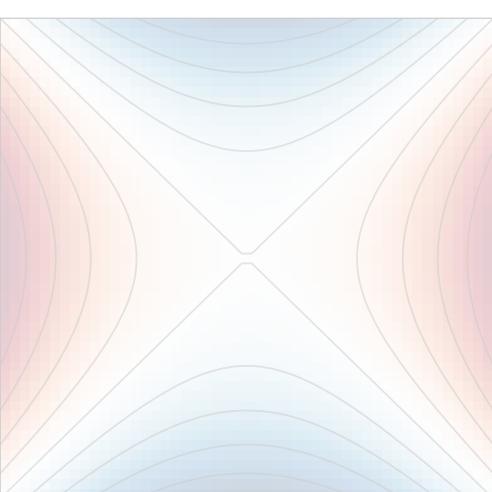
Gradient flow is too simple of a continuous model for SGD.  
It fails to account for key building blocks of modern optimization:

- weight decay
- momentum
- stochasticity
- finite learning rates

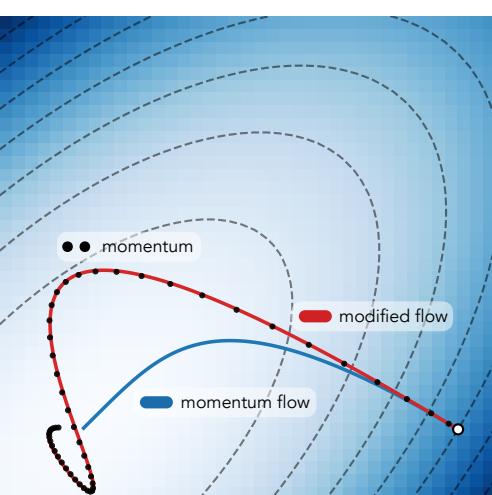
# Q. Can we solve for complex learning dynamics of real deep learning models?



Part 1. Symmetry in the Loss Constrain Gradient and Hessian Geometries



Part 2. Symmetry Leads to Conservation Laws Under Gradient Flow



Part 3. A Realistic Continuous Model for Stochastic Gradient Descent



Part 4. Combining Symmetry and Modified Flow to Derive Learning Dynamics

## Gradient flow is too simple, how can we construct a realistic continuous model for SGD?

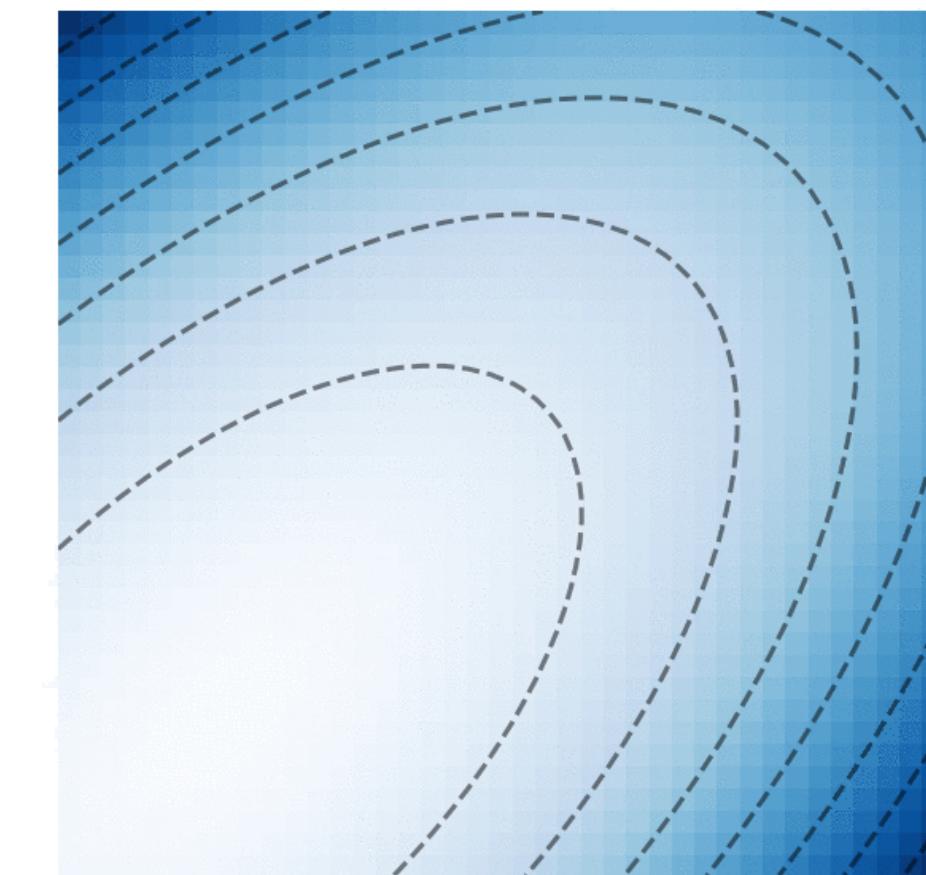
**Gradient Flow:**  $\frac{d\theta}{dt} = -g(\theta)$

**Modeling weight decay ( $\lambda$ ):** Weight decay changes the trajectory from gradient flow pulling the network to the origin in parameter space.

$$\frac{d\theta}{dt} = -g(\theta) - \lambda\theta$$

**Example:** Quadratic Loss

$$\mathcal{L} = \theta^\top A\theta$$



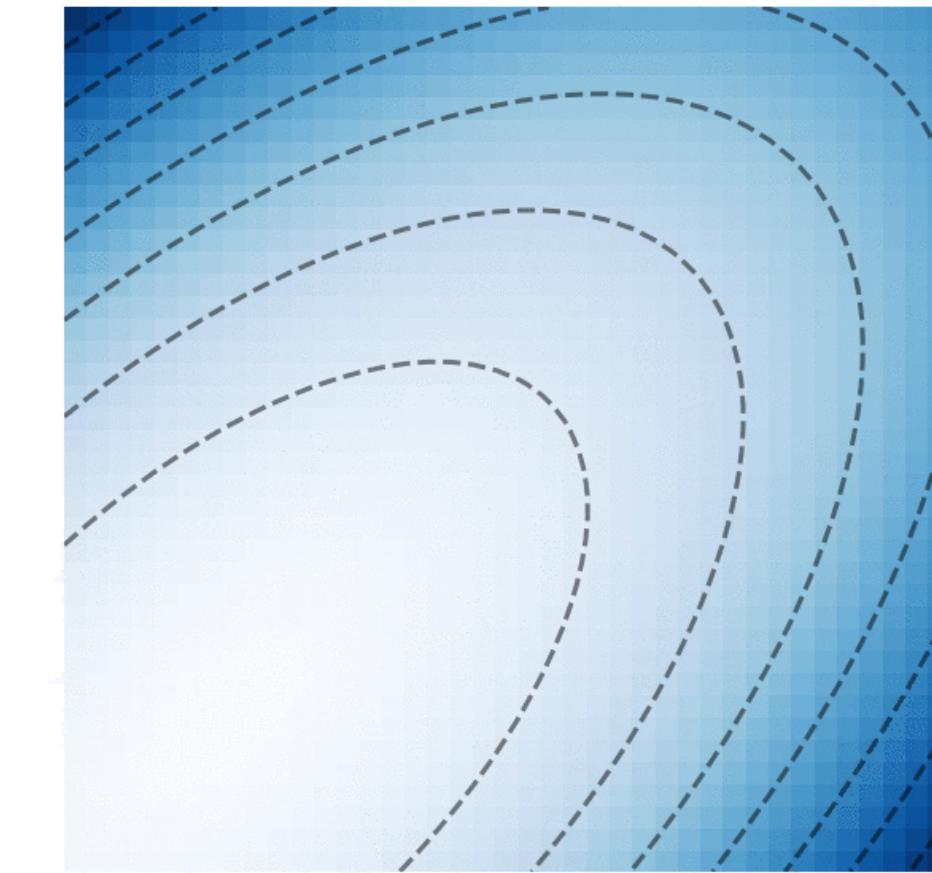
**Modeling momentum ( $\beta$ ):** Momentum accelerates the learning dynamics rescaling time, but leaves the trajectory intact.

$$(1 - \beta)\frac{d\theta}{dt} = -g(\theta)$$

**Modeling stochasticity:** We model the batch gradient  $\hat{g}_{\mathcal{B}}(\theta)$  as a noisy version of the full batch gradient  $g(\theta)$  such that,

$$\hat{g}_{\mathcal{B}}(\theta) = g(\theta) + \epsilon$$

where  $E[\epsilon] = 0$  and  $\langle \hat{g}_{\mathcal{B}}, \partial_\alpha \psi \rangle = \langle \epsilon, \partial_\alpha \psi \rangle = 0$  for any batch  $\mathcal{B}$ .



**Blue curve:** gradient flow

**Red curve:** modified trajectory

# A Realistic Continuous Model for Stochastic Gradient Descent

**Modeling discretization:** Gradient descent moves in the direction of steepest descent, but due to a finite learning rate fails to remain on the continuous steepest descent path.

**Q.** Does there exist a “continuous equation of learning” that can accurately model the effect of a finite learning rate?

**A.** Modified equation analysis is a method for modeling the discrepancy introduced by a discretization of a PDE with higher order “spatial” or “temporal” derivatives.

**Modified Loss:** Introduces higher order derivatives of the loss, effectively modifying the loss landscape itself.

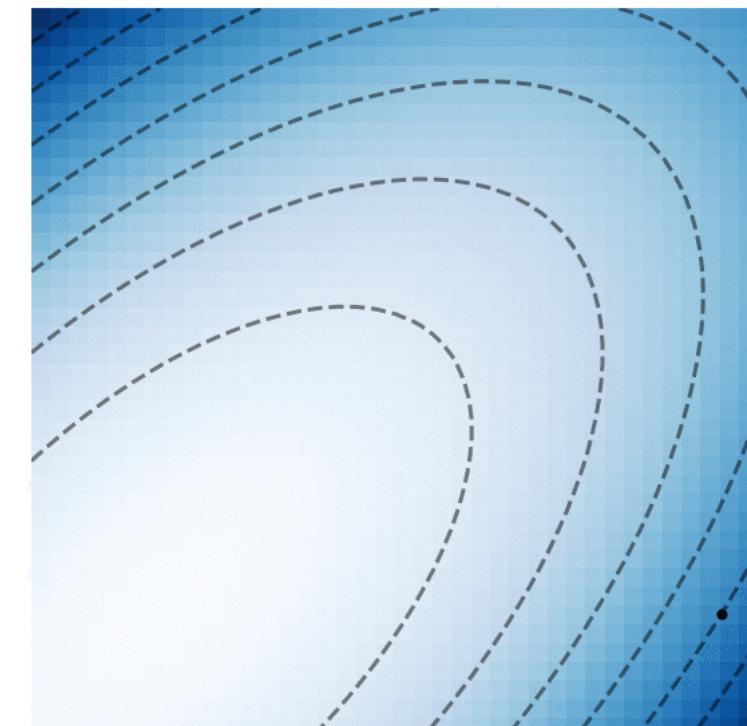
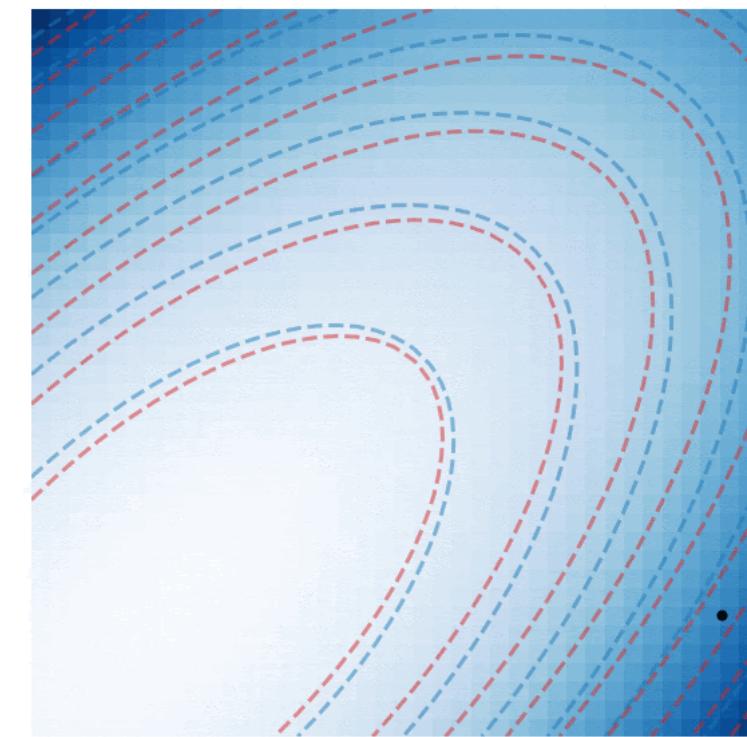
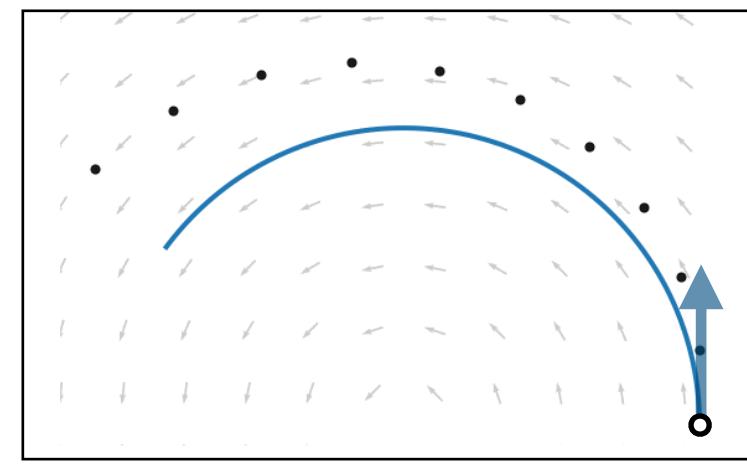
$$\frac{d\theta}{dt} = -g(\theta) - \frac{\eta}{2} H(\theta) g(\theta)$$

David G.T. Barrett and Benoit Dherin. *Implicit Gradient Regularization*. 2020.

**Modified Flow:** Introduces higher order temporal derivatives modifying the flow directly.

$$\frac{d\theta}{dt} = -g(\theta) - \frac{\eta}{2} \frac{d^2\theta}{dt^2}$$

Nikola B. Kovachki, Andrew M. Stuart. *Analysis Of Momentum Methods*. 2019.

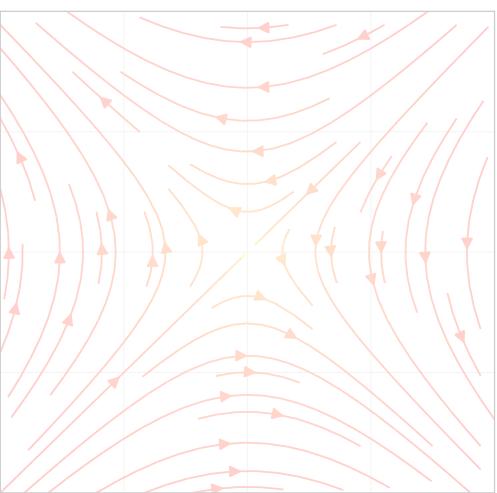


**Blue curve:** gradient flow

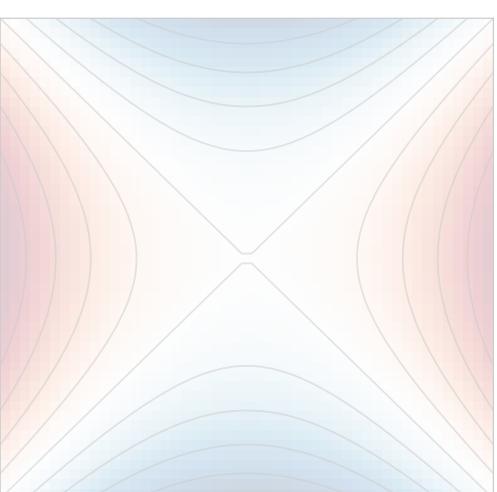
**Red curve:** modified trajectory

**Black dots:** discrete SGD steps

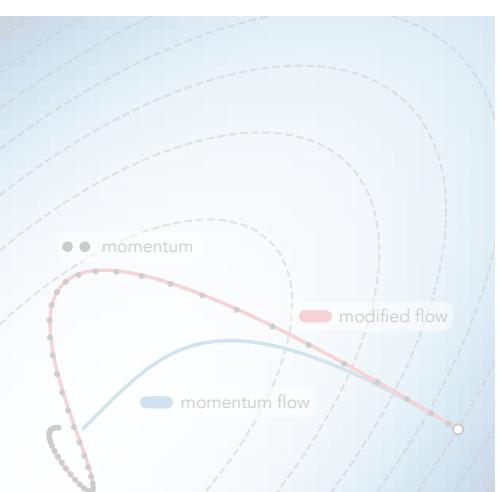
# Q. Can we solve for complex learning dynamics of real deep learning models?



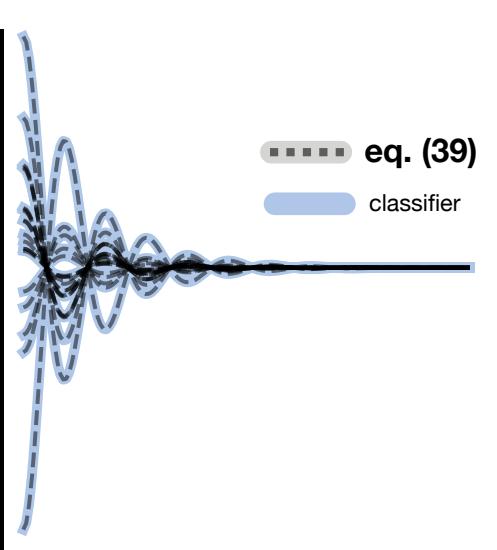
Part 1. Symmetry in the Loss Constrain Gradient and Hessian Geometries



Part 2. Symmetry Leads to Conservation Laws Under Gradient Flow



Part 3. A Realistic Continuous Model for Stochastic Gradient Descent

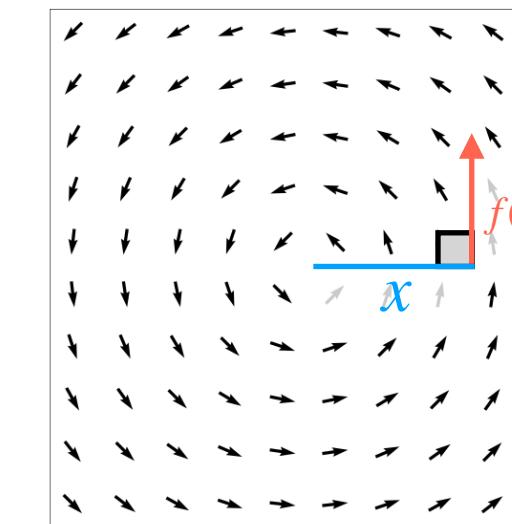


Part 4. Combining Symmetry and Modified Flow to Derive Learning Dynamics

# Q. How do weight decay, momentum, stochastic gradients, and finite learning rates all interact to break these conservation laws?

**Example:** Consider the circular vector field on  $\mathbb{R}^2$ :

$$f(x) = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} x$$



$$\langle f(x), x \rangle = 0$$

1). Equation of learning.

**Discrete**

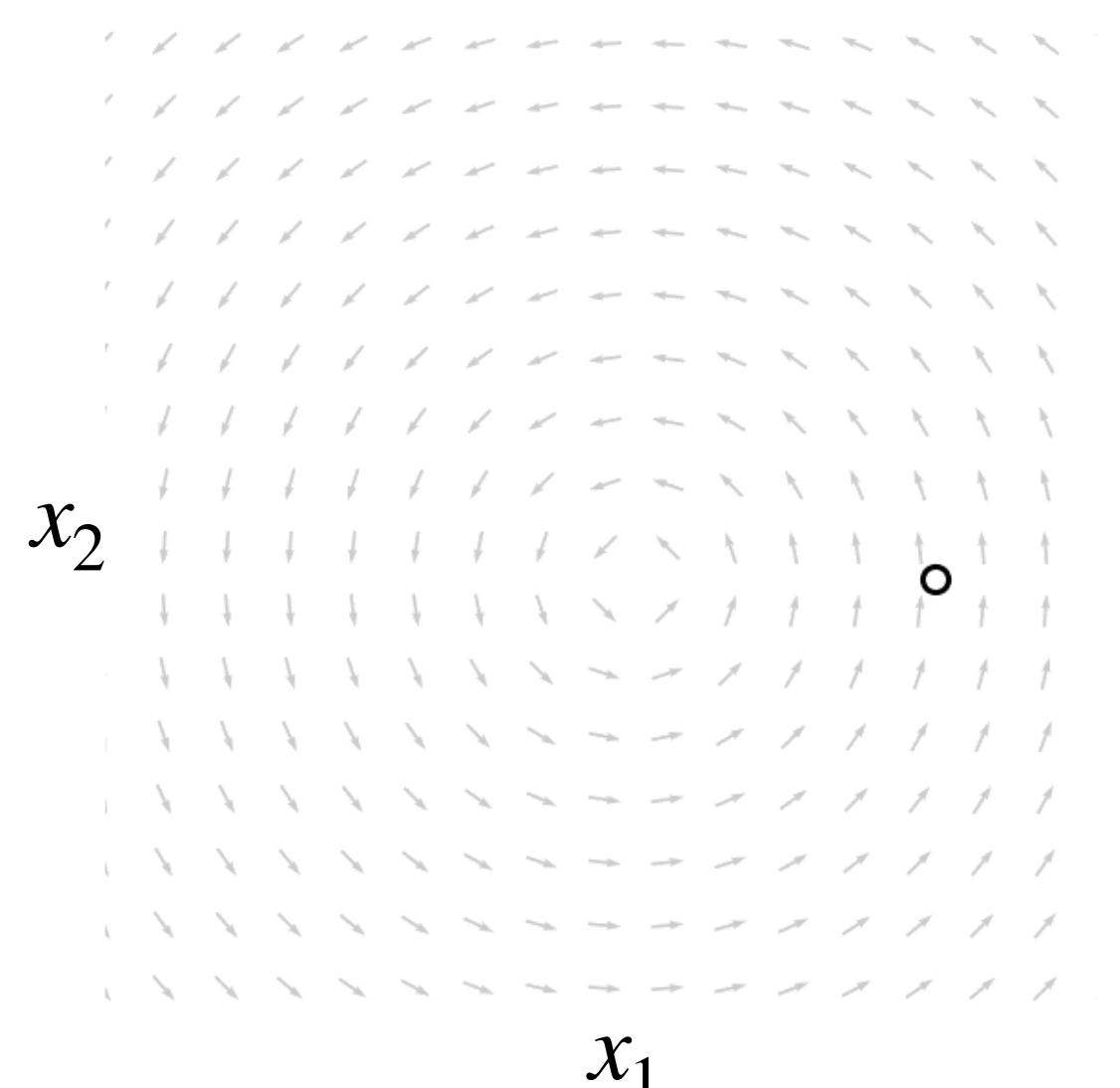
$$\bullet \bullet \quad x_{t+1} = x_t + \eta f(x_t)$$

**Circular flow**

$$\textcolor{blue}{\frac{dx}{dt}} = f(x)$$

**Modified circular flow**

$$\textcolor{red}{\frac{dx}{dt}} = f(x) + \frac{\eta}{2} x$$



2). Project the learning dynamics onto the generator vector fields associated with a symmetry.

3). Harness the **geometric constraints** introduced by symmetry to derive simplified ODEs.

$$\langle f(x), x \rangle = 0$$

$$\langle \frac{dx}{dt}, x \rangle = \cancel{\langle f(x), x \rangle}$$

$$\langle f(x), x \rangle = 0$$

$$\langle \frac{dx}{dt}, x \rangle = \cancel{\langle f(x), x \rangle} + \langle \frac{\eta}{2} x, x \rangle$$

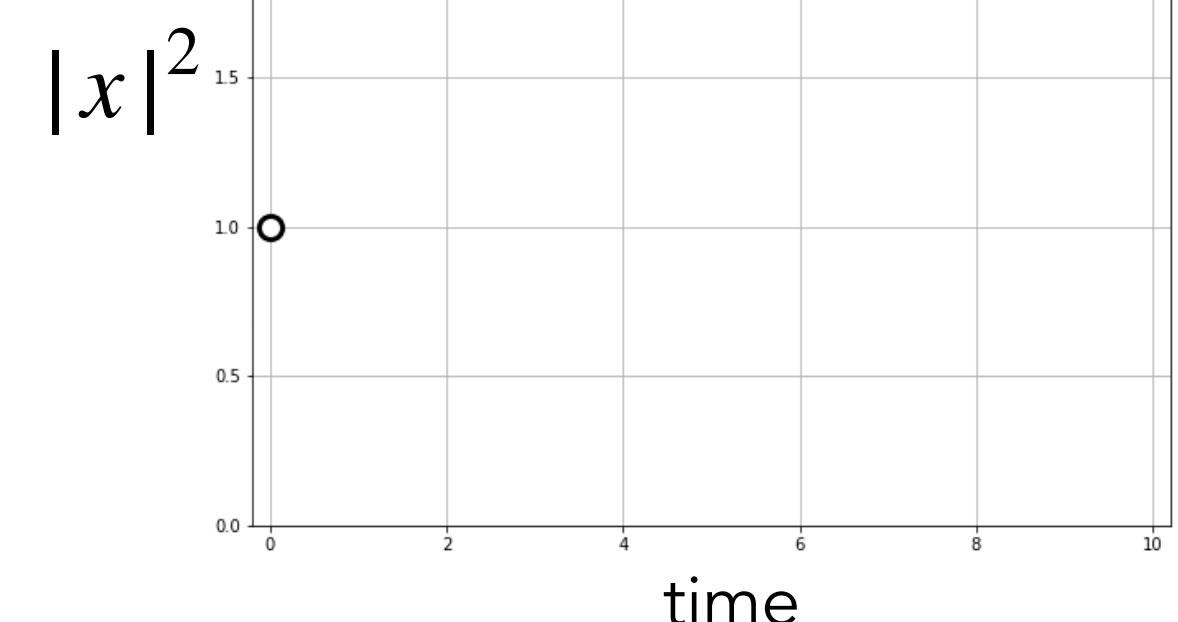
4). Solve these ODEs to obtain exact dynamics for the previously conserved quantities.

$$\textcolor{blue}{\frac{d}{dt} \frac{1}{2} |x|^2} = 0$$

$$|x|^2 = |x(0)|^2$$

$$\textcolor{red}{\frac{d}{dt} \frac{1}{2} |x|^2} = \frac{\eta}{2} |x|^2$$

$$|x|^2 = e^{\eta t} |x(0)|^2$$



# Q. How do weight decay, momentum, stochastic gradients, and finite learning rates all interact to break these conservation laws?

To answer, we:

1. Consider a realistic continuous model for SGD, an equation of learning.

$$d\theta = -\tilde{F}(\theta)dt + \sqrt{\frac{\eta}{S}}G(\theta)dW_t, \quad \tilde{F}(\theta) = (1 + \lambda)g + \lambda\theta + \frac{\eta}{2}(Hg + H\theta).$$

2. Project the learning dynamics onto the generator vector fields associated with a symmetry.
3. Harness the **geometric constraints** introduced by symmetry to derive simplified ODEs.

$$\langle \tilde{F}, \theta \rangle = (1 + \lambda)\cancel{\langle g, \theta \rangle} + \lambda\langle \theta, \theta \rangle + \frac{\eta}{2}(\cancel{\langle Hg, \theta \rangle} + \cancel{\langle H\theta, \theta \rangle}) = \lambda|\theta|^2 - \frac{\eta}{2}|g|^2$$

4. Solve these ODEs to obtain exact dynamics for the previously conserved quantities.

$$\frac{d|\theta|^2}{dt} = -\lambda|\theta|^2 + \frac{\eta}{2}|g|^2$$

Translation

$$\langle \theta_{\mathcal{A}}(t), \mathbb{1} \rangle = e^{-\lambda t} \langle \theta_{\mathcal{A}}(0), \mathbb{1} \rangle$$

Scale

$$|\theta_{\mathcal{A}}(t)|^2 = e^{-2\lambda t} |\theta_{\mathcal{A}}(0)|^2 + \eta \int_0^t e^{-2\lambda(t-\tau)} |g_{\mathcal{A}}|^2 d\tau$$

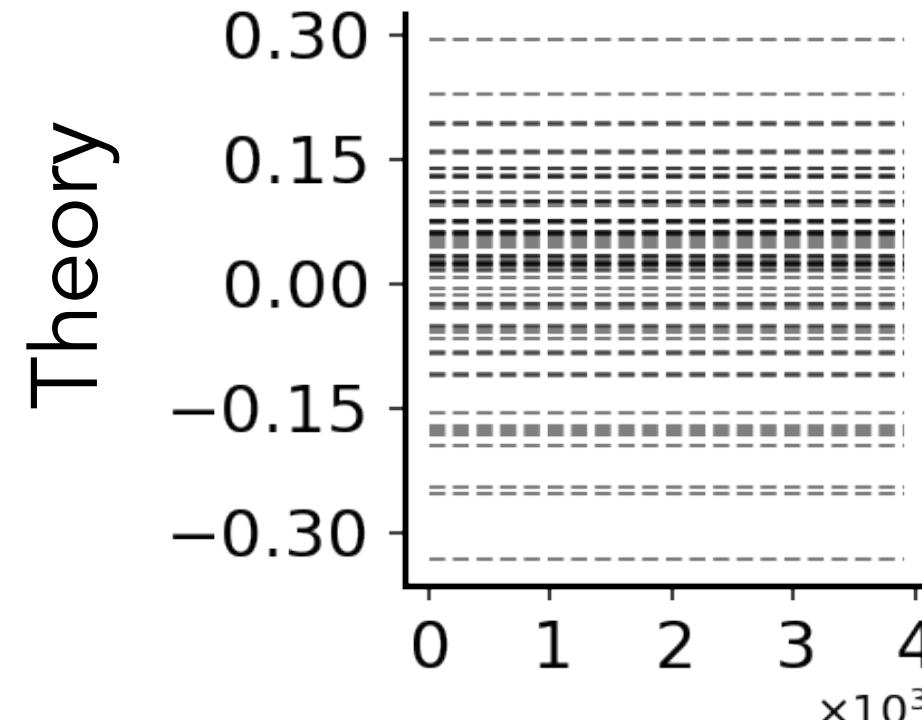
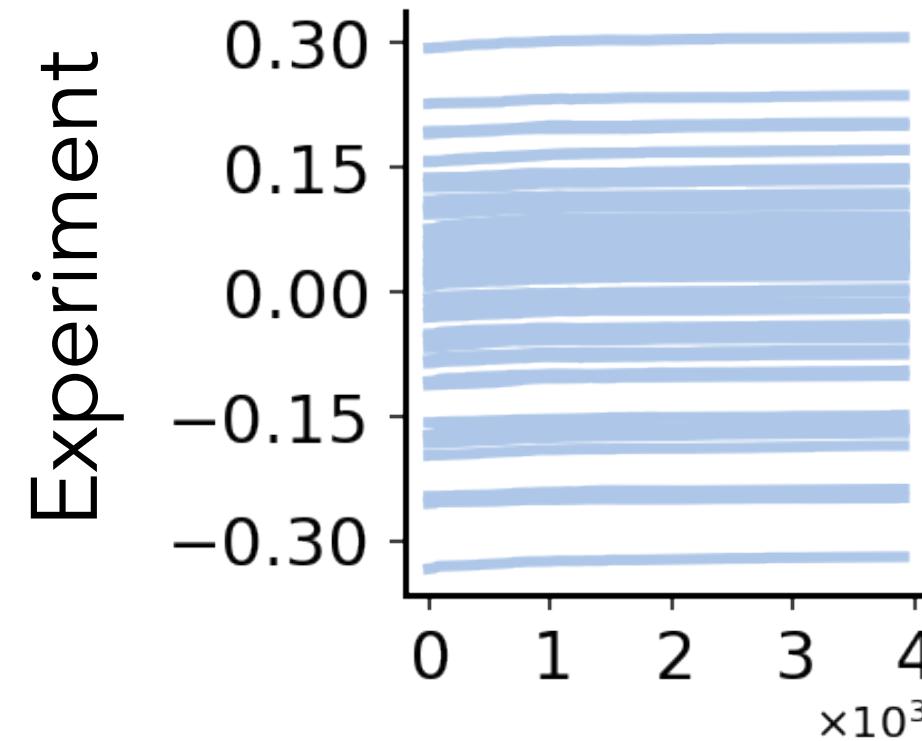
Rescale

$$\begin{aligned} |\theta_{\mathcal{A}_1}(t)|^2 - |\theta_{\mathcal{A}_2}(t)|^2 &= \\ e^{-2\lambda t}(|\theta_{\mathcal{A}_1}(0)|^2 - |\theta_{\mathcal{A}_2}(0)|^2) + \eta \int_0^t e^{-2\lambda(t-\tau)} \left( |g_{\theta_{\mathcal{A}_1}}|^2 - |g_{\theta_{\mathcal{A}_2}}|^2 \right) d\tau &= \end{aligned}$$

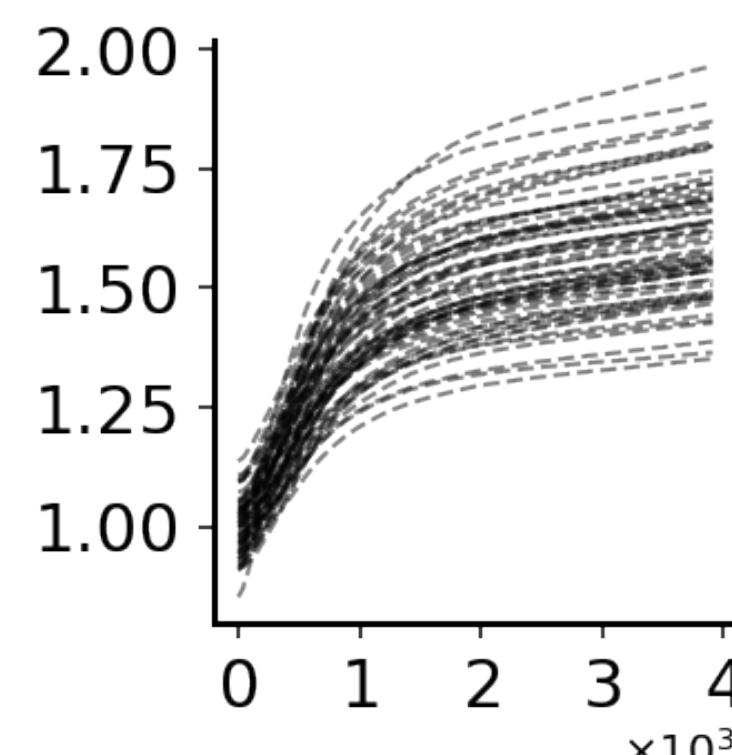
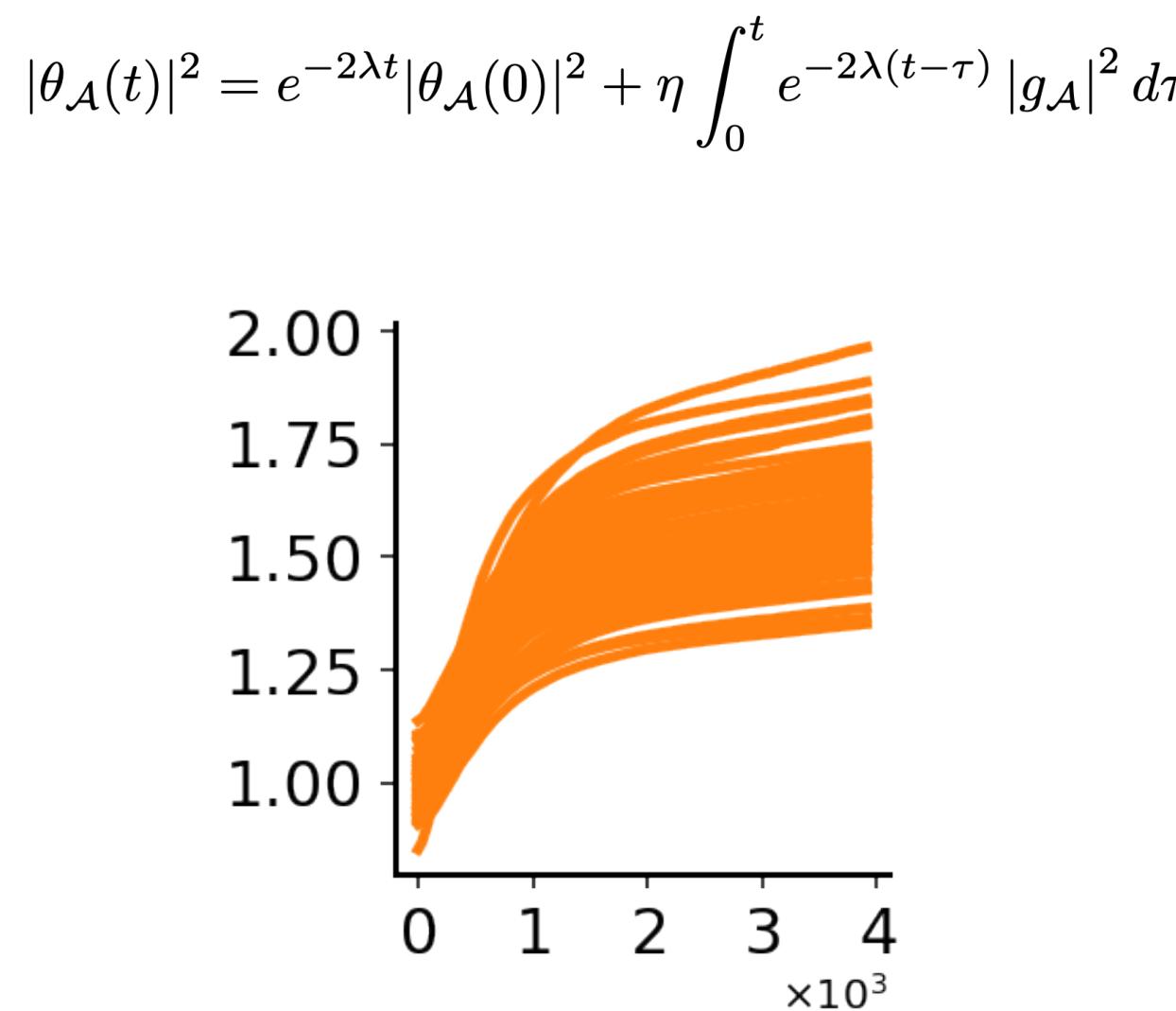
Theory (dotted lines) match the empirics (colored lines) perfectly!

VGG-16 trained on Tiny ImageNet with SGD

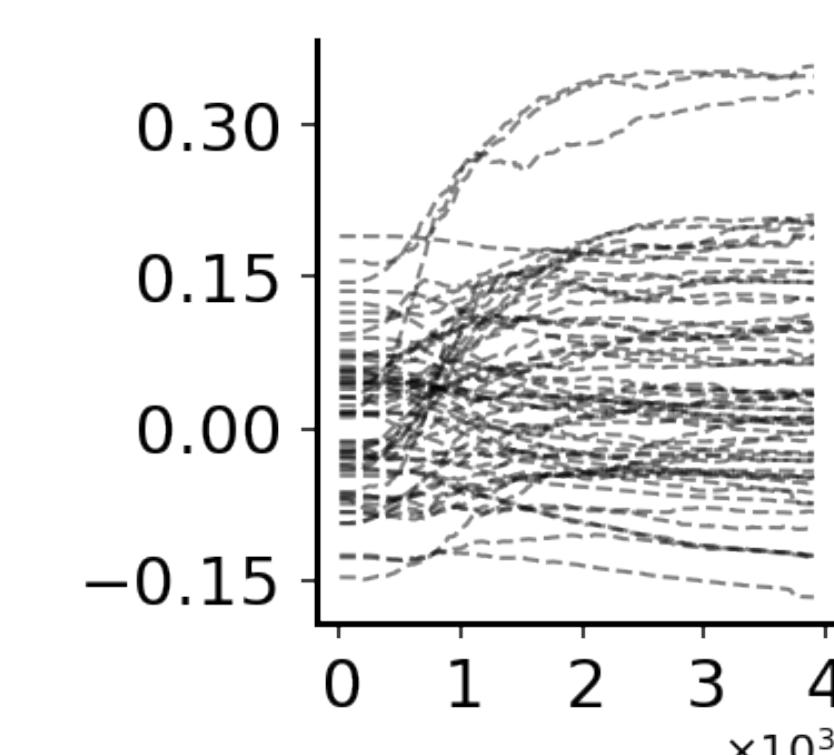
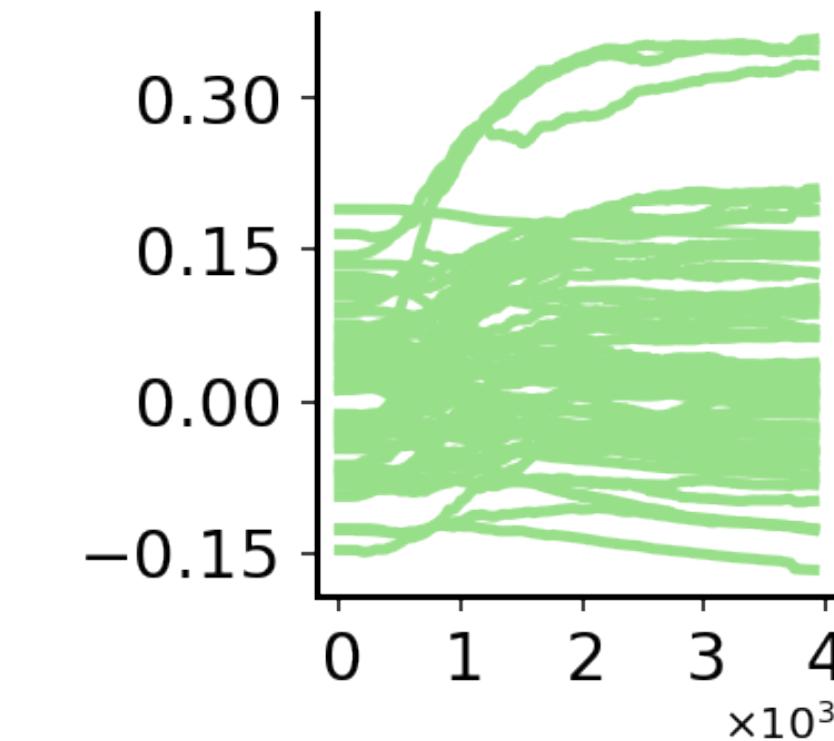
Translation



Scale



Rescale

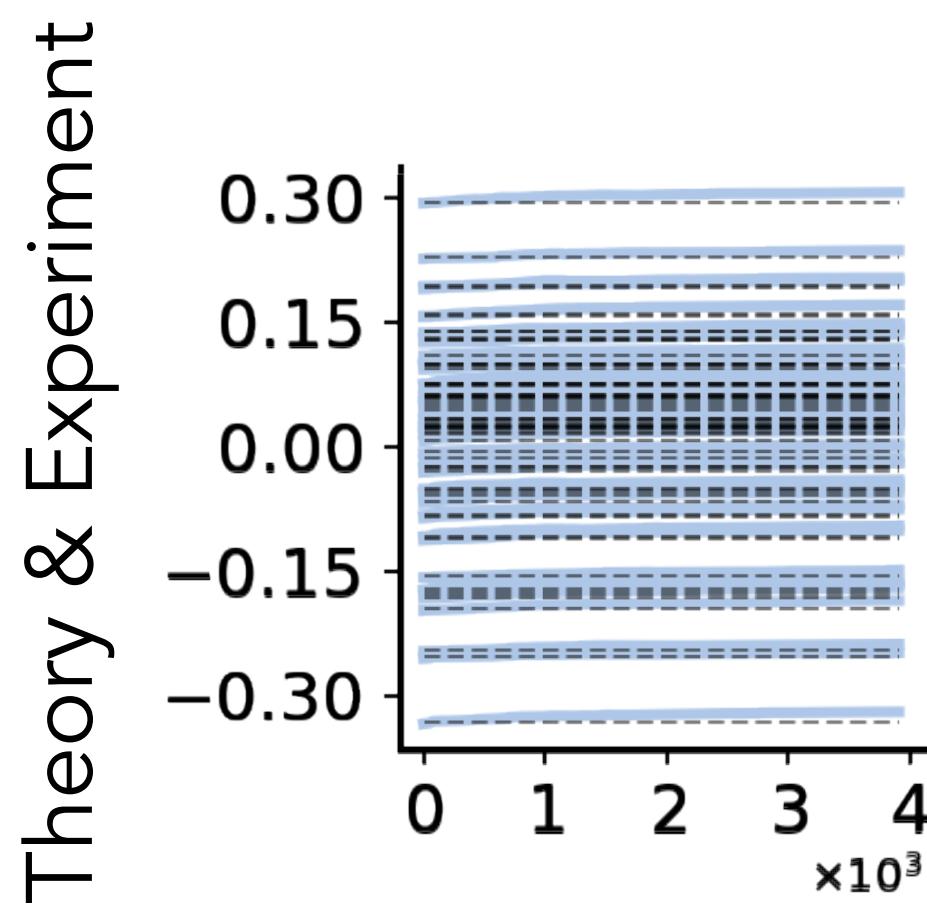


Theory (dotted lines) match the empirics (colored lines) perfectly!

VGG-16 trained on Tiny ImageNet with SGD

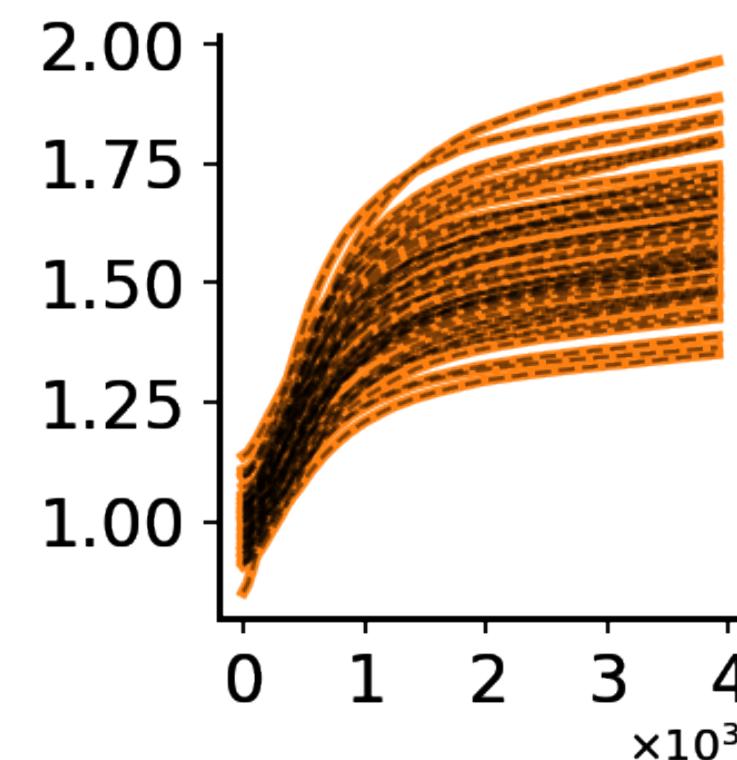
Translation

$$\langle \theta_{\mathcal{A}}(t), \mathbb{1} \rangle = e^{-\lambda t} \langle \theta_{\mathcal{A}}(0), \mathbb{1} \rangle$$



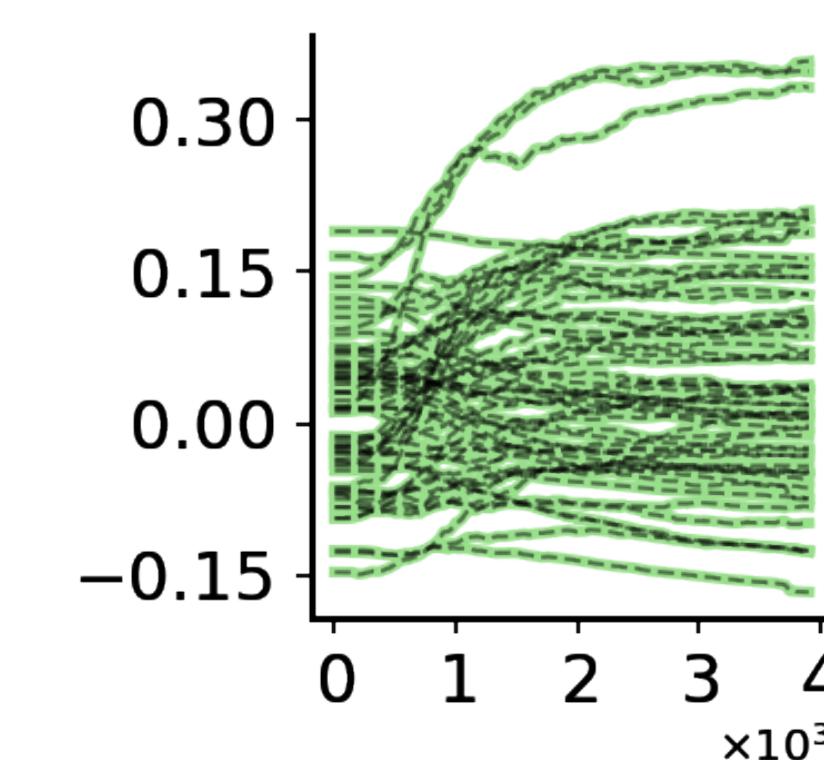
Scale

$$|\theta_{\mathcal{A}}(t)|^2 = e^{-2\lambda t} |\theta_{\mathcal{A}}(0)|^2 + \eta \int_0^t e^{-2\lambda(t-\tau)} |g_{\mathcal{A}}|^2 d\tau$$



Rescale

$$\begin{aligned} |\theta_{\mathcal{A}_1}(t)|^2 - |\theta_{\mathcal{A}_2}(t)|^2 &= \\ e^{-2\lambda t} (|\theta_{\mathcal{A}_1}(0)|^2 - |\theta_{\mathcal{A}_2}(0)|^2) + \eta \int_0^t e^{-2\lambda(t-\tau)} \left( |g_{\theta_{\mathcal{A}_1}}|^2 - |g_{\theta_{\mathcal{A}_2}}|^2 \right) d\tau & \end{aligned}$$

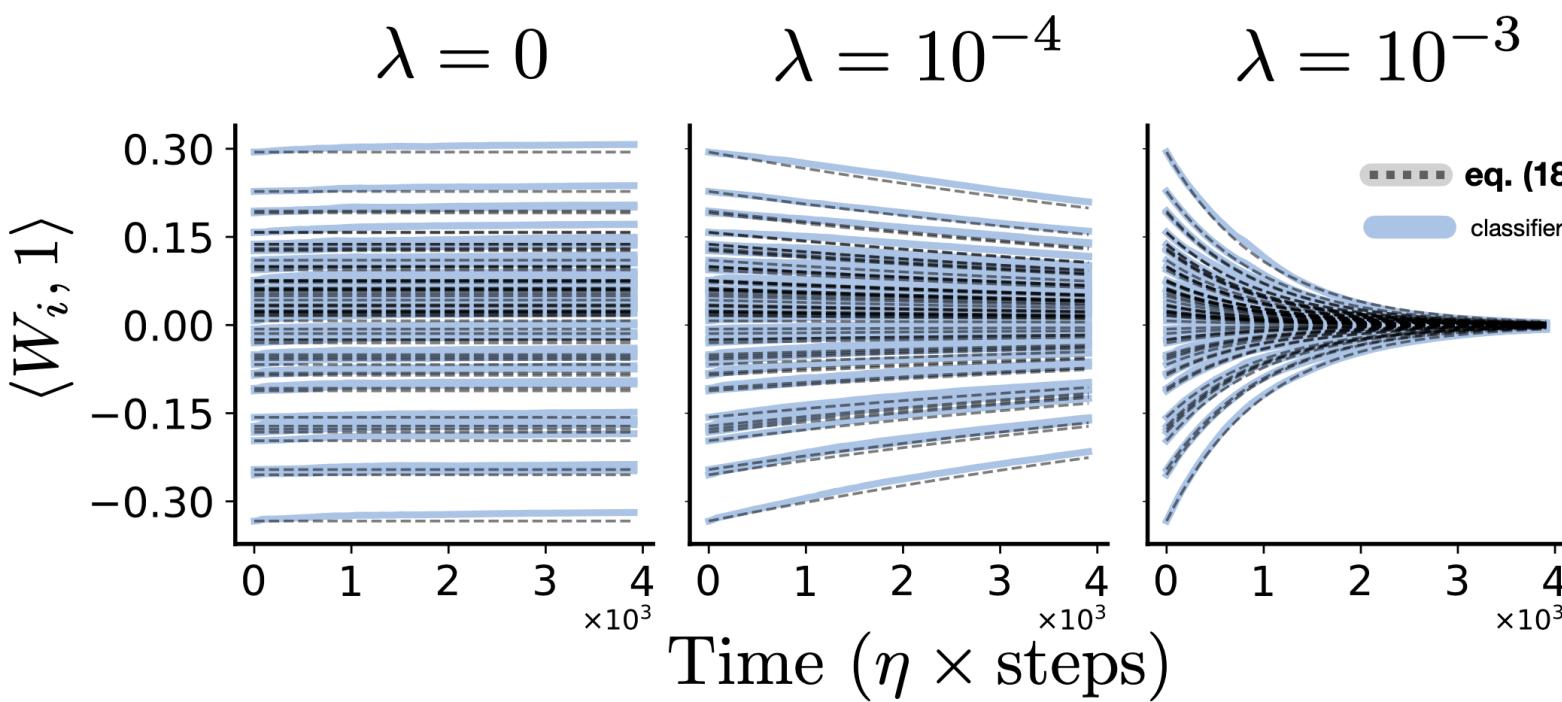


# Theory (dotted lines) match the empirics (colored lines) perfectly!

VGG-16 trained on Tiny ImageNet with SGD

## Translation

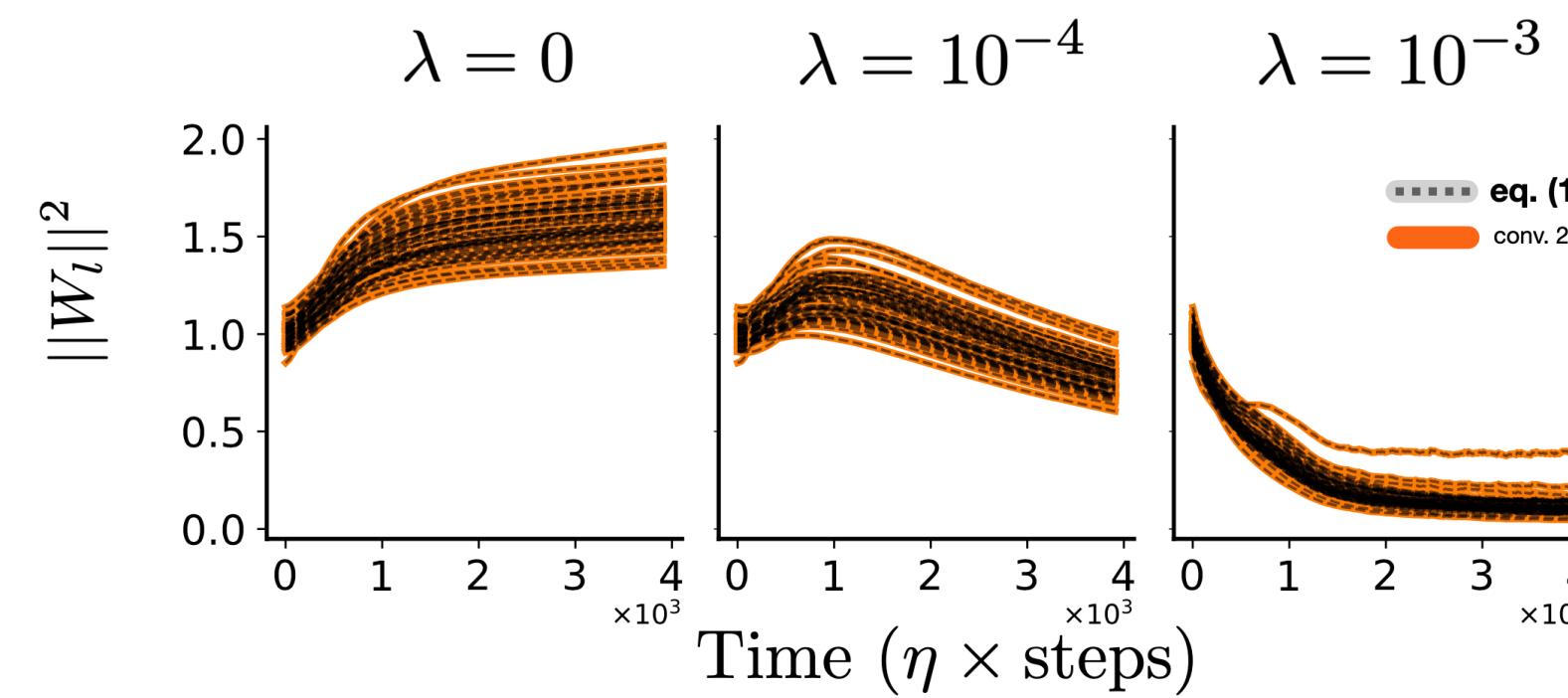
$$\langle \theta_{\mathcal{A}}(t), \mathbb{1} \rangle = e^{-\lambda t} \langle \theta_{\mathcal{A}}(0), \mathbb{1} \rangle$$



- $\langle \theta_{\mathcal{A}}(t), 1 \rangle$  decays exponentially to zero at a rate proportional to the weight decay.
- Dynamics is independent of learning rate and data due to the lack of curvature in the gradient field

## Scale

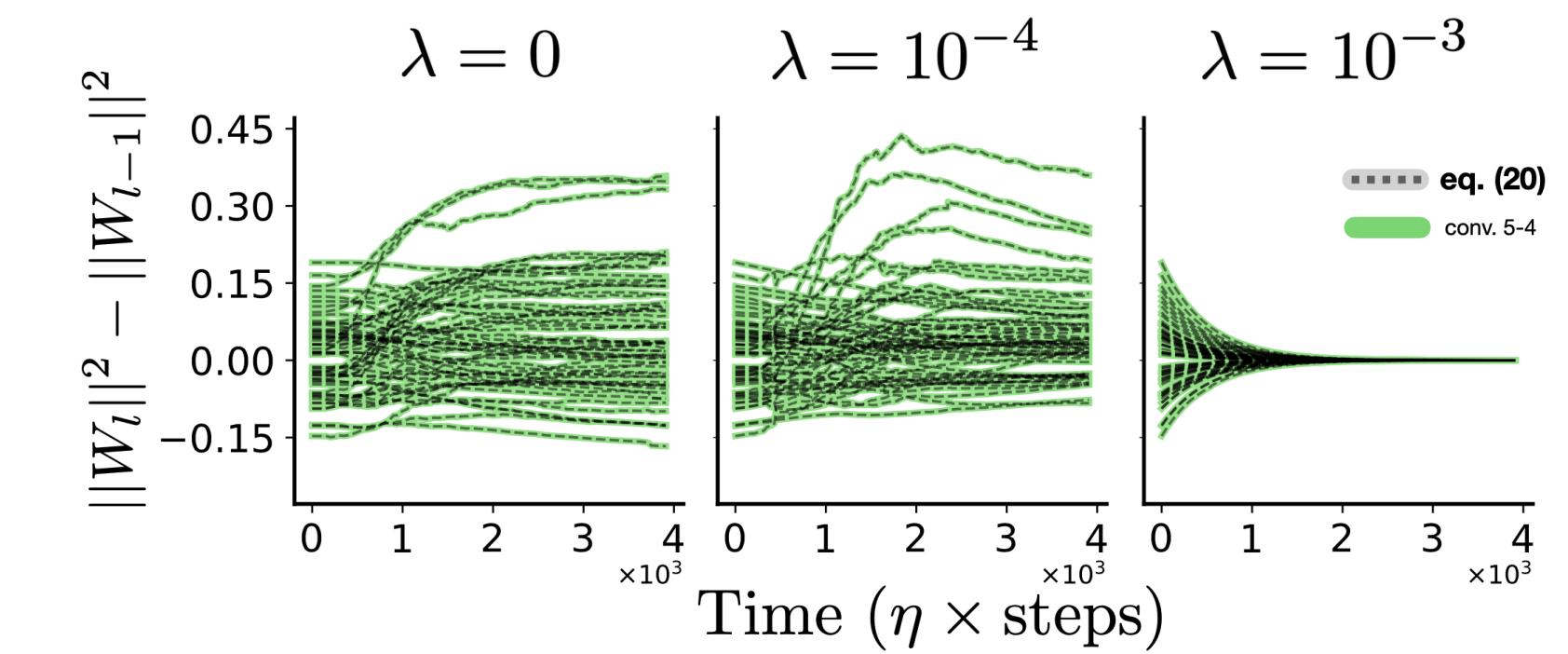
$$|\theta_{\mathcal{A}}(t)|^2 = e^{-2\lambda t} |\theta_{\mathcal{A}}(0)|^2 + \eta \int_0^t e^{-2\lambda(t-\tau)} |g_{\mathcal{A}}|^2 d\tau$$



- Norm  $|\theta_{\mathcal{A}}|^2$  is the sum of an exponentially decaying memory of the norm at initialization and an exponentially weighted integral of gradient norms accumulated through training.

## Rescale

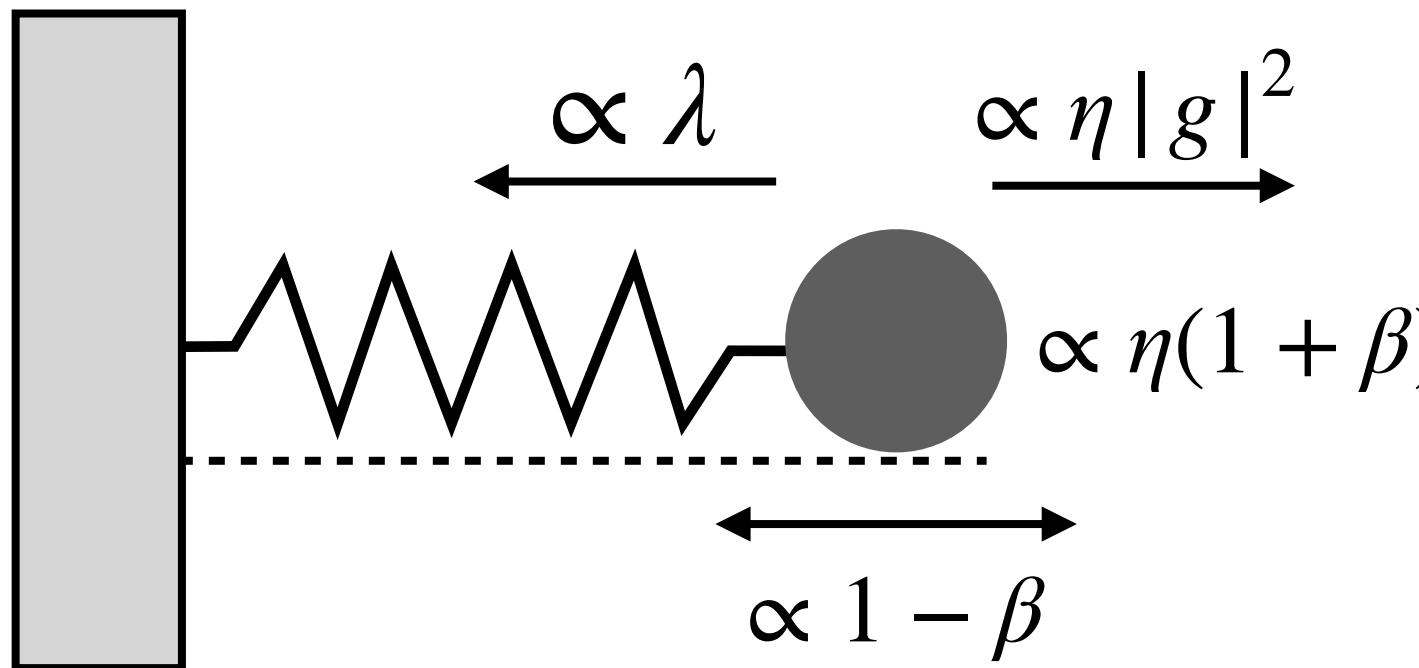
$$|\theta_{\mathcal{A}_1}(t)|^2 - |\theta_{\mathcal{A}_2}(t)|^2 = e^{-2\lambda t} (|\theta_{\mathcal{A}_1}(0)|^2 - |\theta_{\mathcal{A}_2}(0)|^2) + \eta \int_0^t e^{-2\lambda(t-\tau)} (|g_{\theta_{\mathcal{A}_1}}|^2 - |g_{\theta_{\mathcal{A}_2}}|^2) d\tau$$



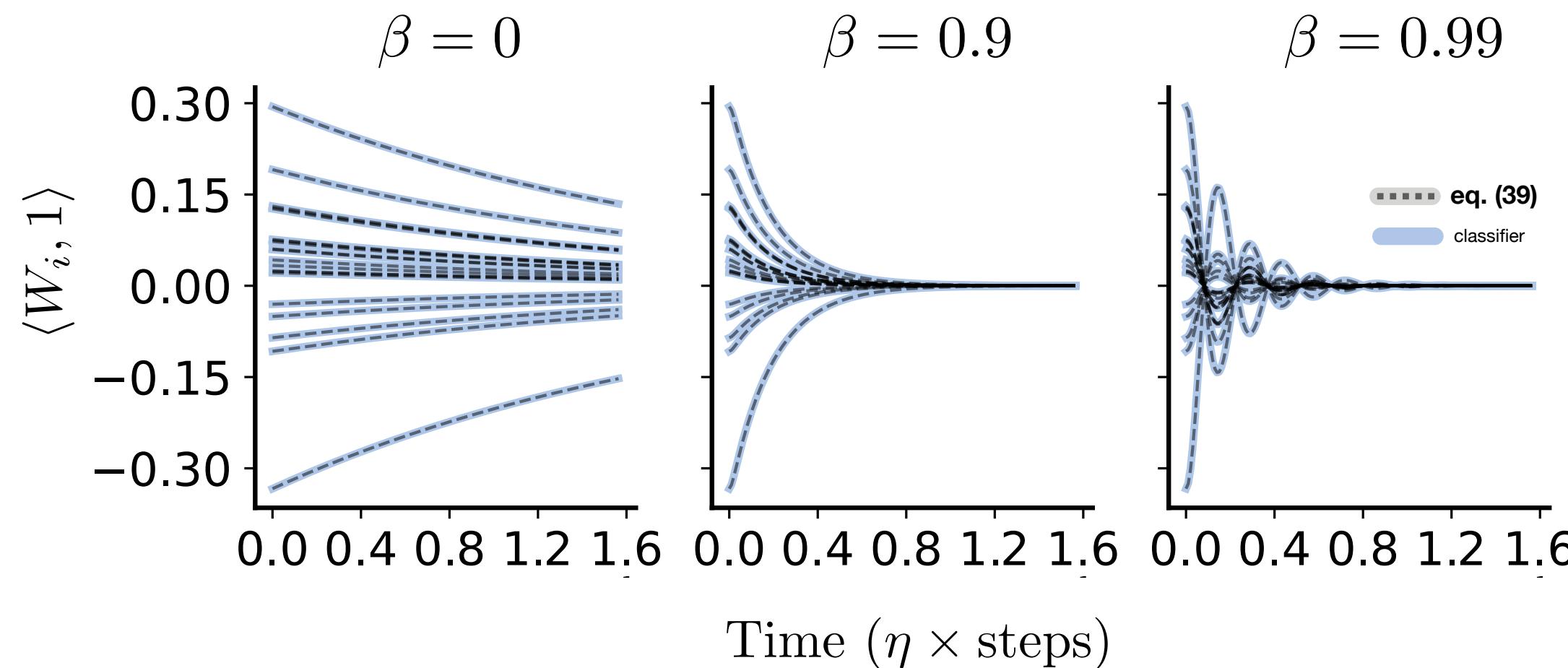
- Similar to the scale dynamics, the rescale dynamics do depend on the data through the gradient norms
- No guarantee that the integral term is always positive.

# Harmonic oscillation with momentum

Harmonic oscillator pulled by regularizer, pushed by gradients, with mass scaling with learning rate!



**Harmonic oscillation with momentum.** When considering the learning dynamics of momentum, the solutions we obtain take the form of driven harmonic oscillators, where the optimization hyperparameters have physical interpretations.



We plot the column sum of the final linear layer of a VGG-16 model (without batch normalization) trained on Tiny ImageNet.

# Conceptual Overview

## Classical Mechanics

v.s.

## Neural Mechanics



**Forces:**

Gravity, Electric/Magnetic, Friction etc...

**Equation of motion:**

Newton's Law ( $F(x) = m\partial_t^2 x$ )

**Symmetries in Lagrangian:**

Translation in time/space, Rotation

**Conservation laws:**

Energy, momentum, angular momentum

**Forces:**

Gradients driven by real world dataset

**Equation of learning:**

Modified gradient flow

**Symmetries in the Loss function:**

Translation, Scale, Rescale

**Broken conservation laws:**

Dynamics of parameter combinations

## Conclusion and Future Work

Two “hammers” developed and used in this work:

1. **Symmetry**: A unifying theoretical framework explaining how a network’s architecture leads to geometric properties in the gradient and Hessian.
2. **Modified Gradient Flow**: A realistic continuous equation modeling SGD with weight decay, momentum, stochasticity, and discretization.

