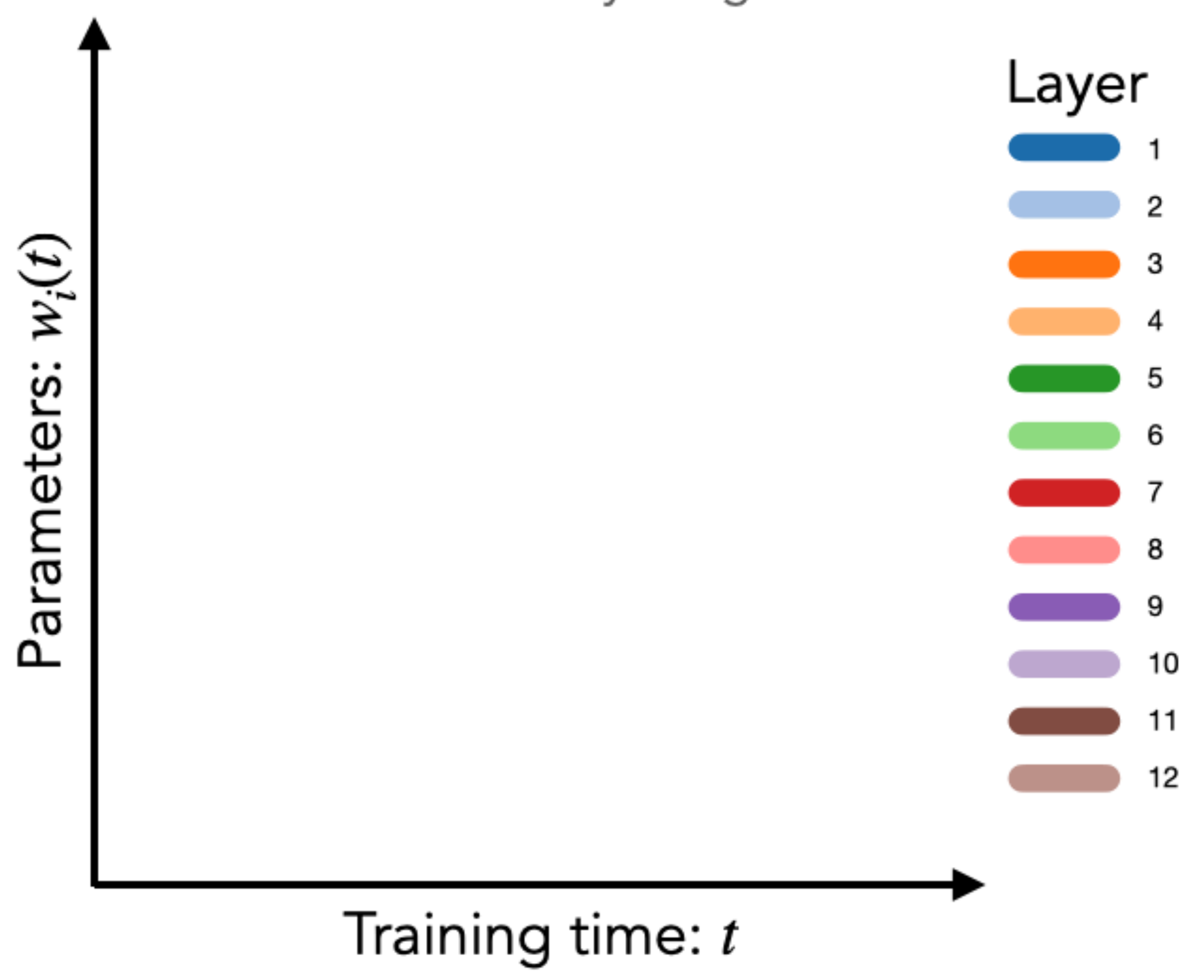
Neural Mechanics:
Symmetry and Broken Conservation Laws in Deep Learning Dynamics

Q. Can we solve for complex learning dynamics?

VGG-16 on Tiny ImageNet



Daniel Kunin*, Javier Sagastuy-Brena, Surya Ganguli, Daniel L.K. Yamins, Hidenori Tanaka*

(* equal contribution)

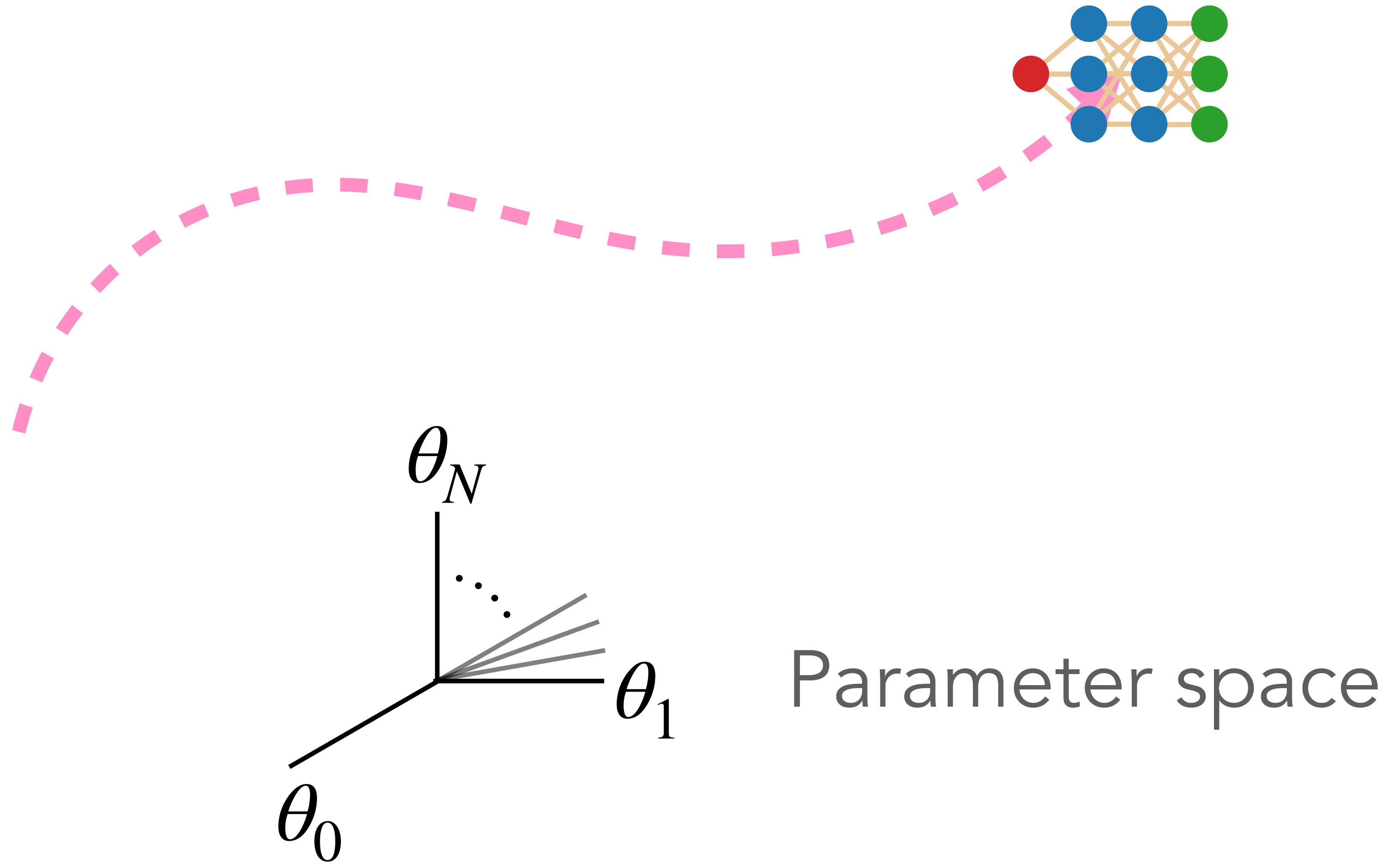
Stanford University, NTT Physics & Informatics Lab

Deep learning has been successful, but it’s inner-workings are still mysterious

There is a myriad of design choices for a deep learning system.
These choices shape the trajectory the network takes during training.

Architecture

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



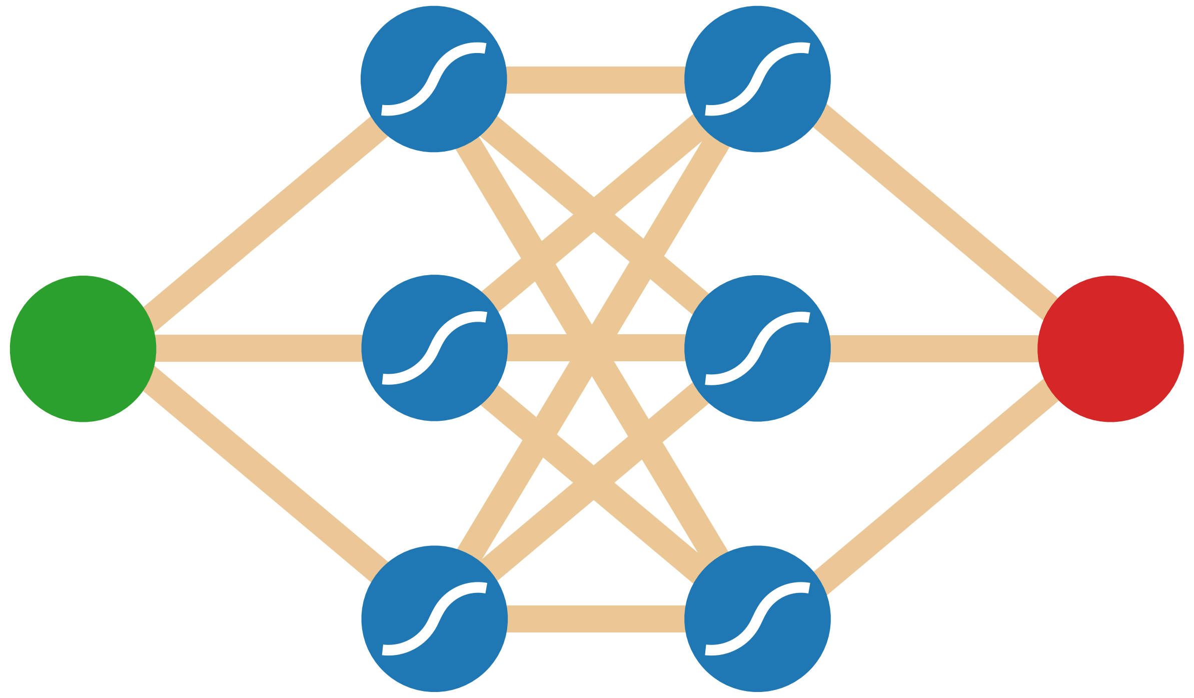
Optimizer

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

Researchers and practitioners largely depend on heuristics and trial & error.
Better understanding of the dynamics is necessary for 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?



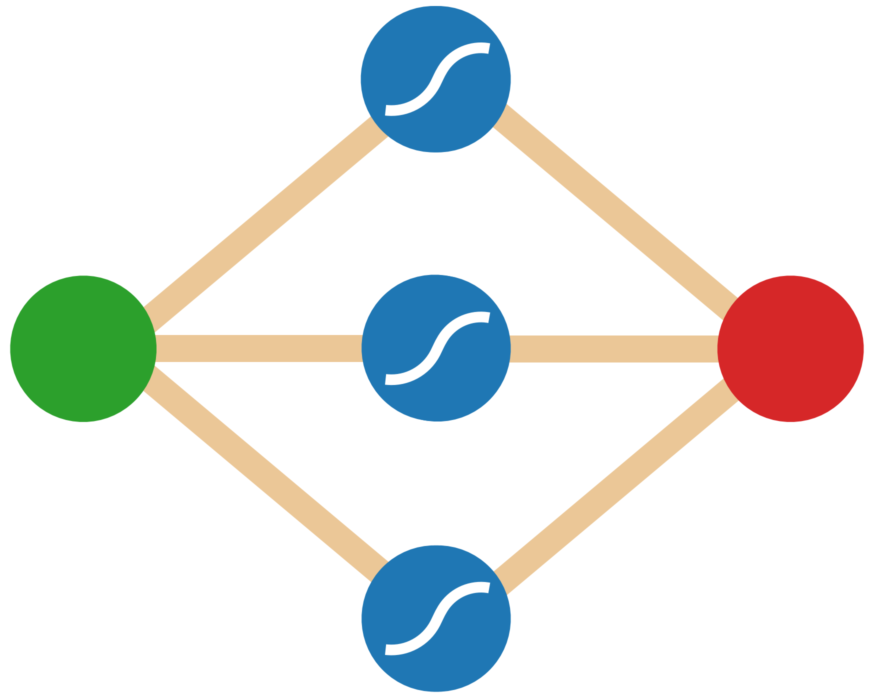
This question is difficult because of…

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

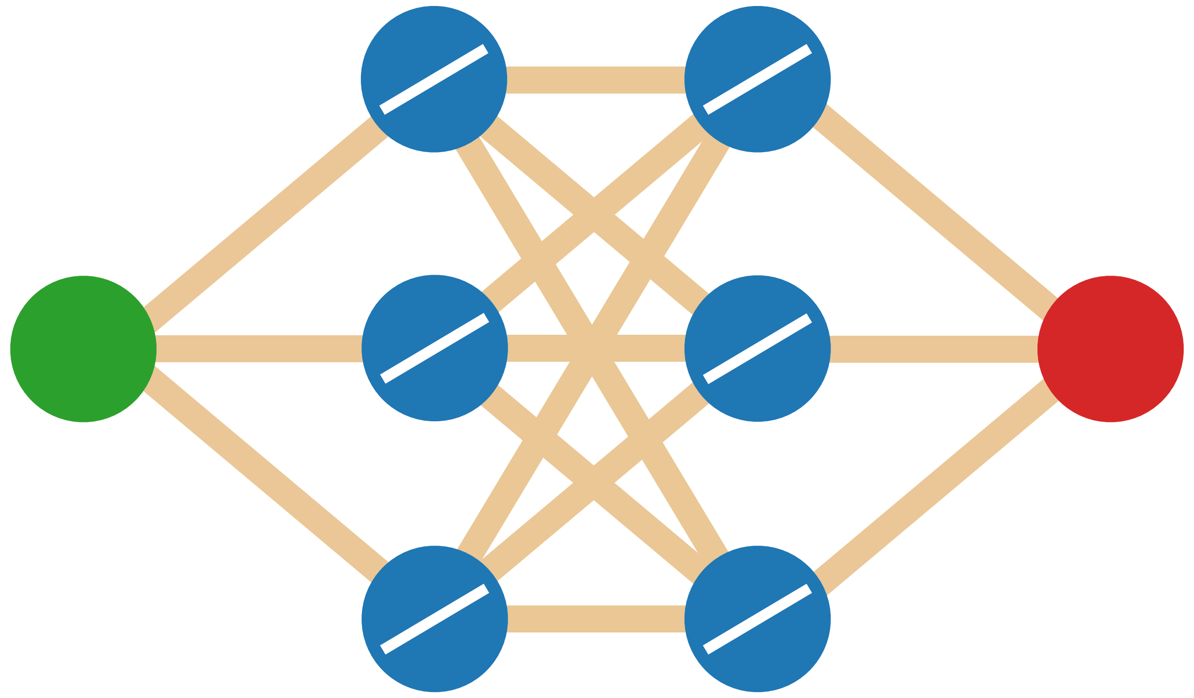
$$y = \theta^{[2]} f(\theta^{[1]} x)$$



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

Linear Networks

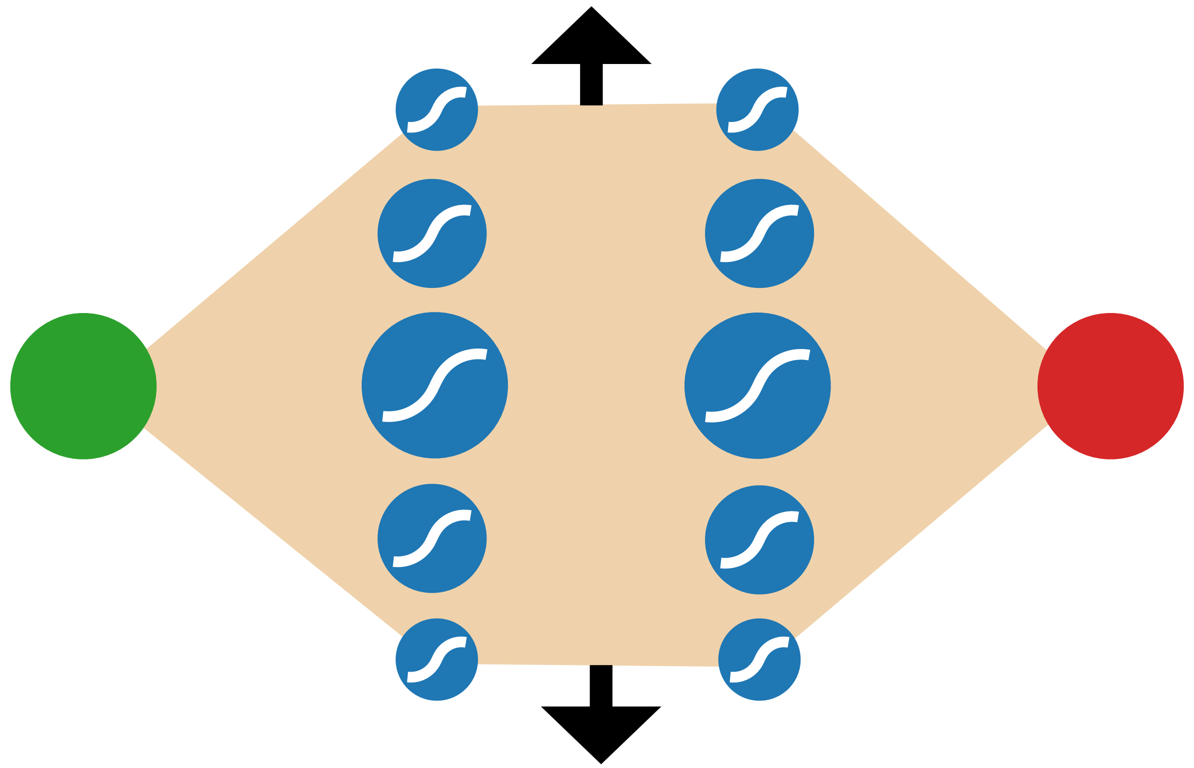
$$y = \theta^{[L]} \dots \theta^{[2]} \theta^{[1]} x$$



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

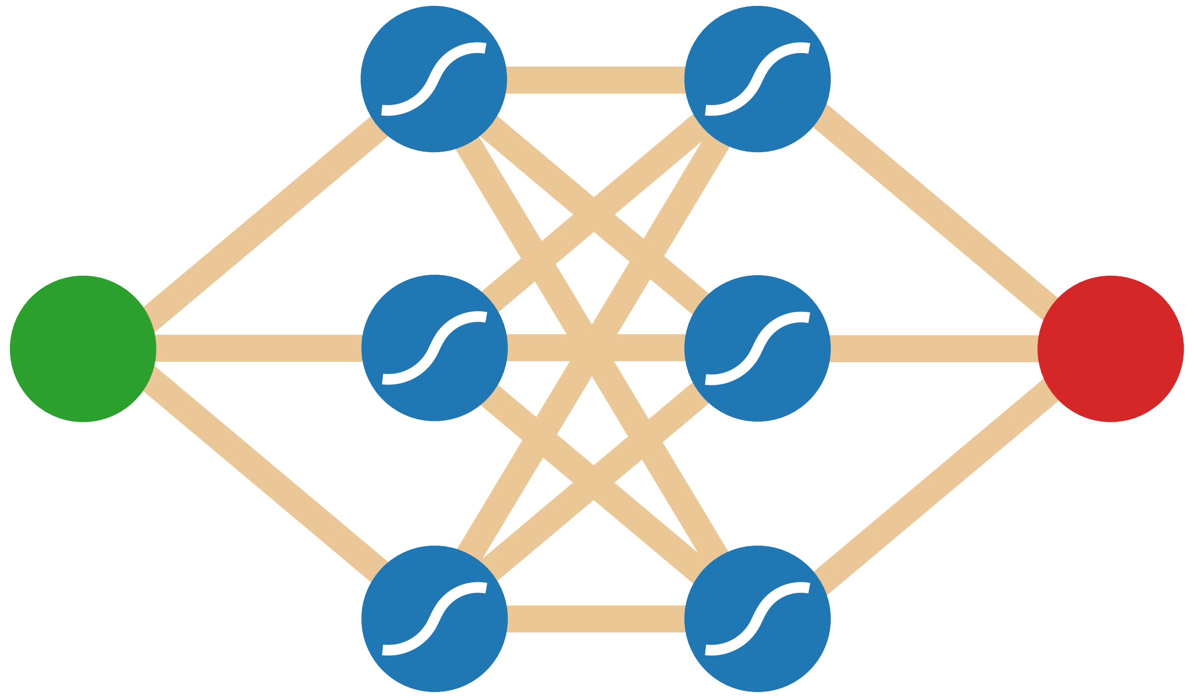
Infinitely Wide

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



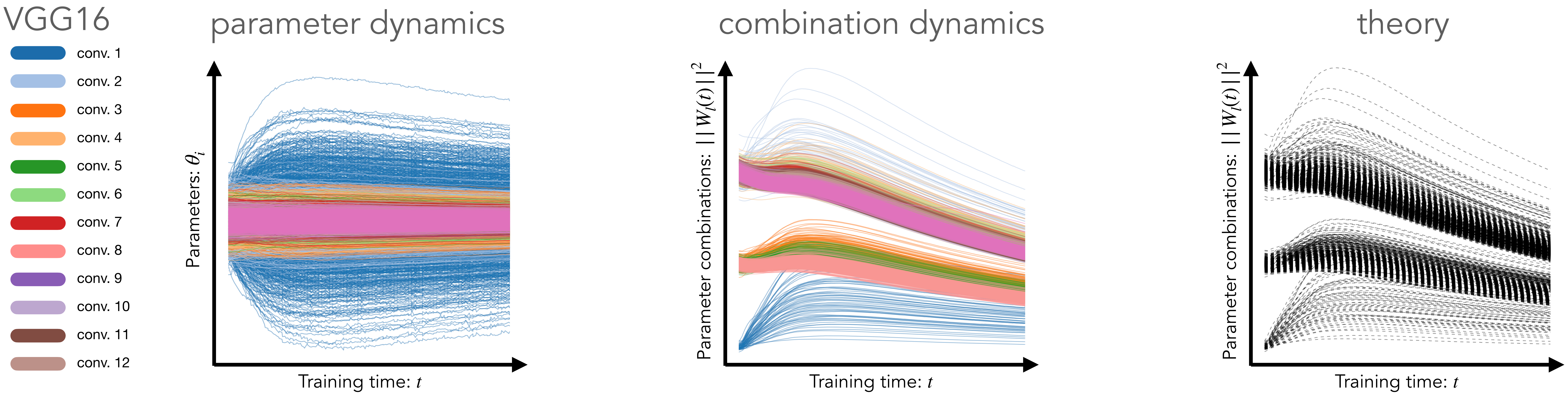
Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. 2018.

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?

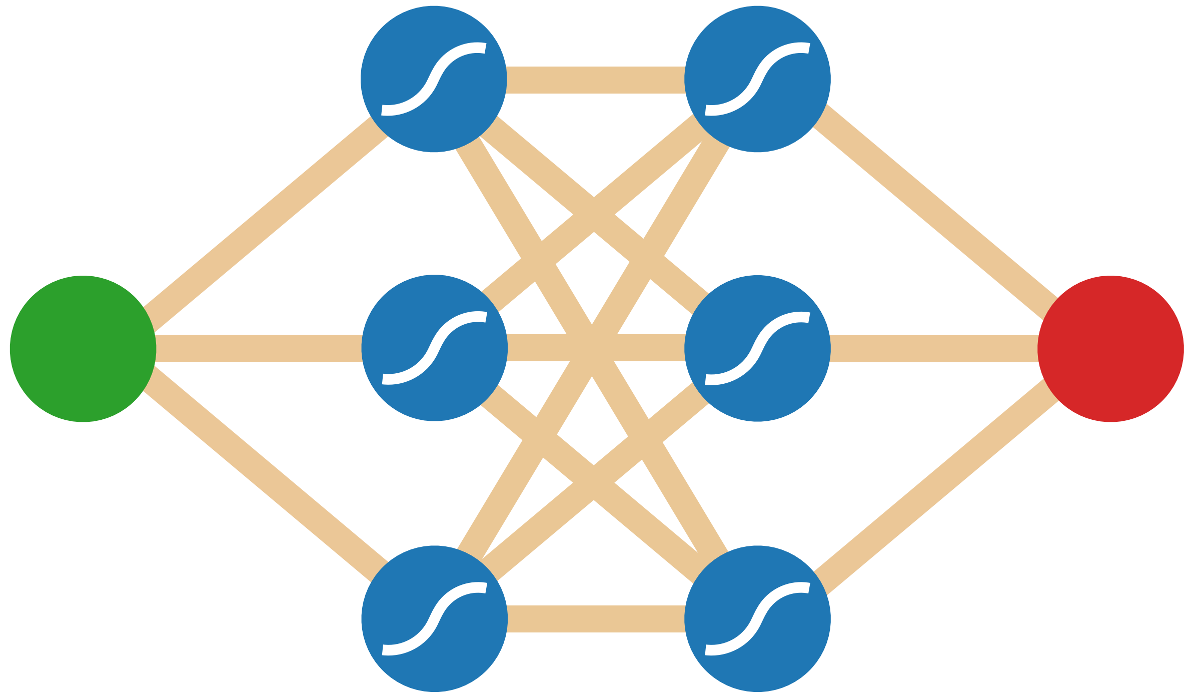


In this work we **don’t introduce major simplifying assumptions** on the architecture or optimizer!

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

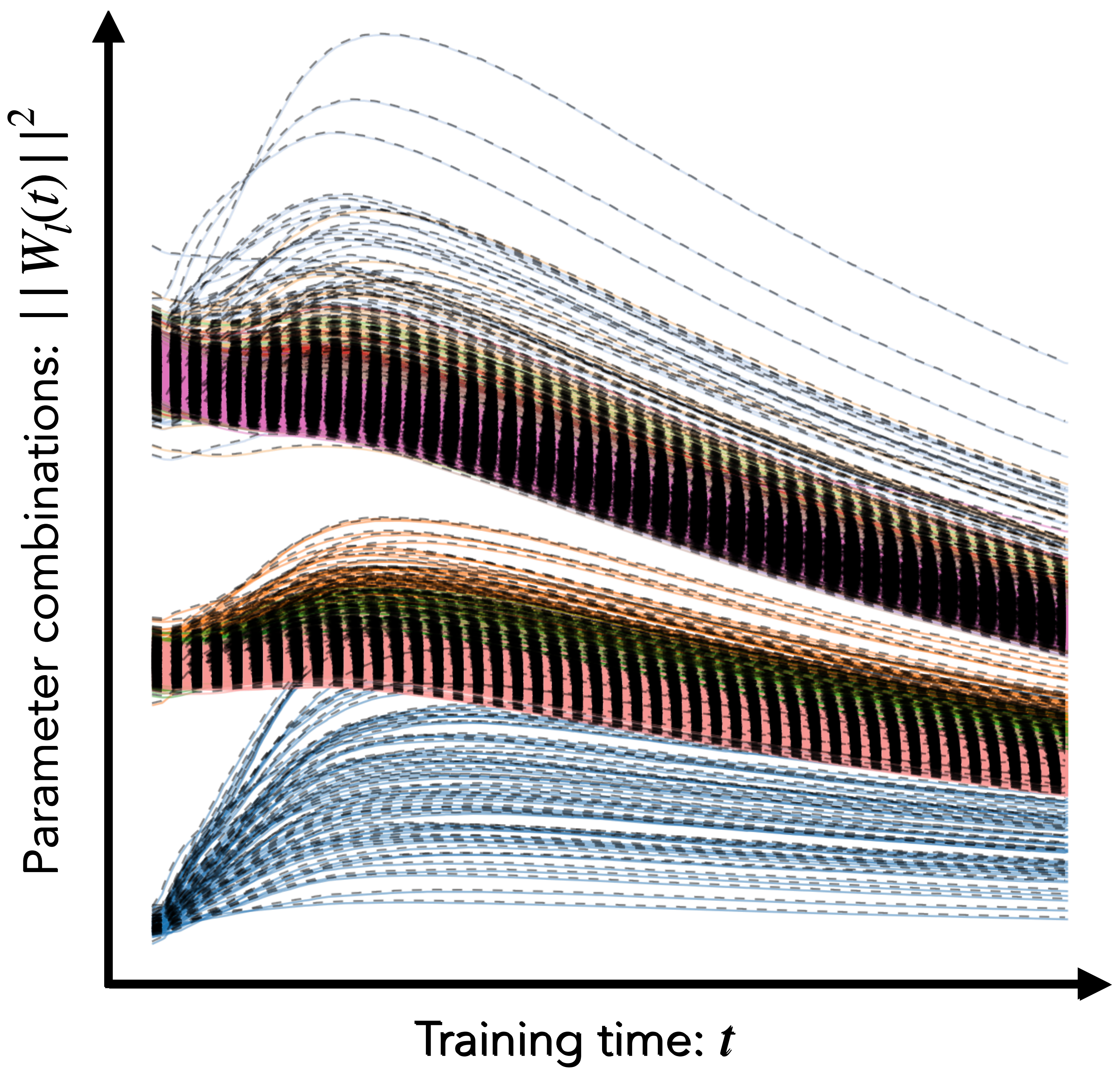


Q. What, if anything, can we quantitatively understand about the learning dynamics of state-of-the-art deep learning models driven by real-world datasets?



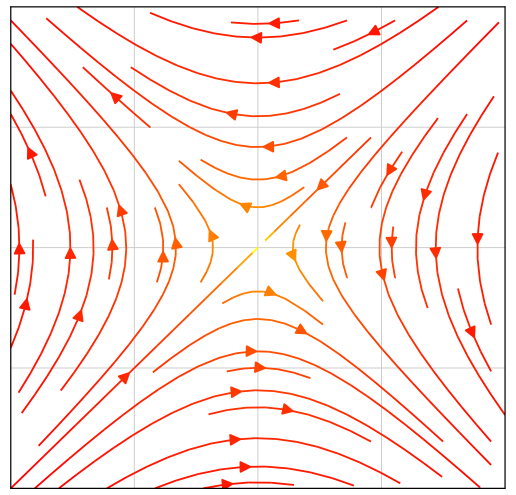
In this work we **don't introduce major simplifying assumptions** on the architecture or optimizer!

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

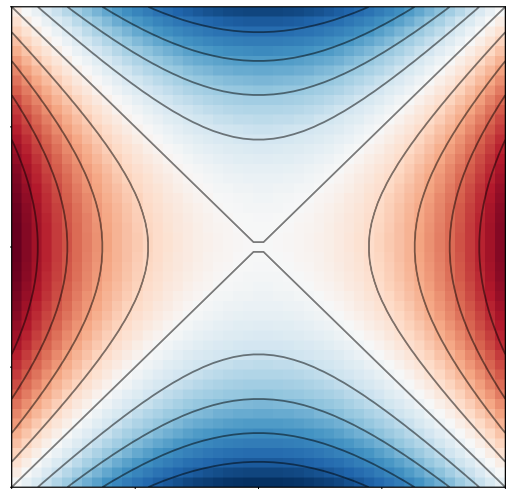


Our theory matches experiment exactly!

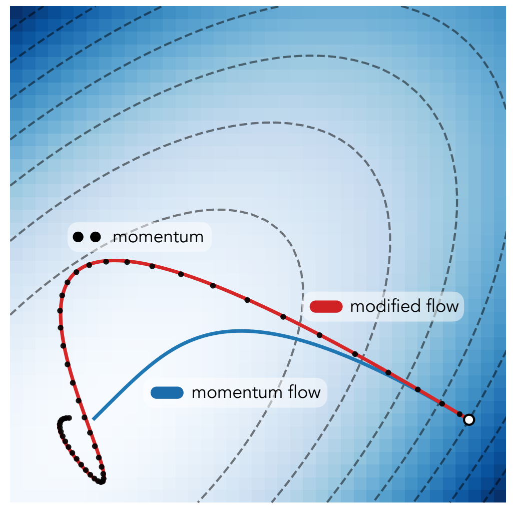
Q. Can we solve for complex learning dynamics of real deep learning models?



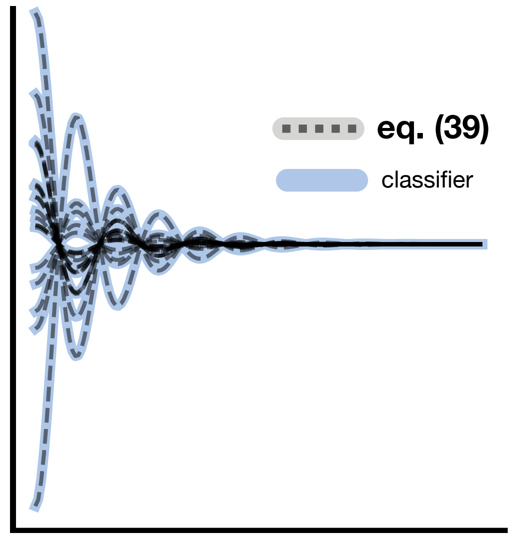
Part 1. Symmetry in the Loss Constrain Gradient and Hessian Geometries



Part 2. Symmetry Leads to Conservation Laws Under Gradient Flow

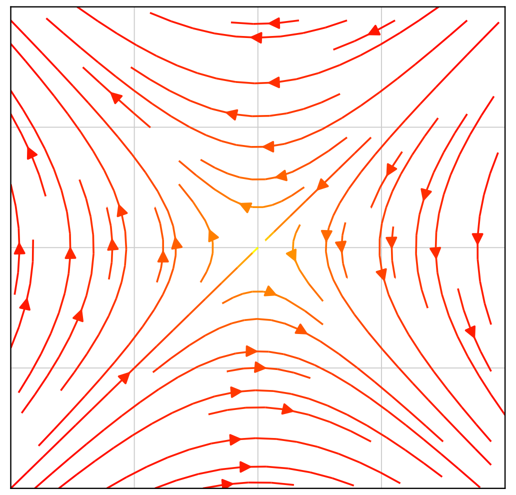


Part 3. A Realistic Continuous Model for Stochastic Gradient Descent

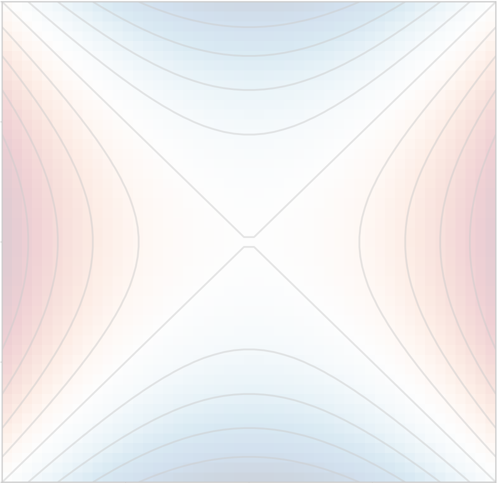


Part 4. Combining Symmetry and Modified Flow to Derive Learning Dynamics

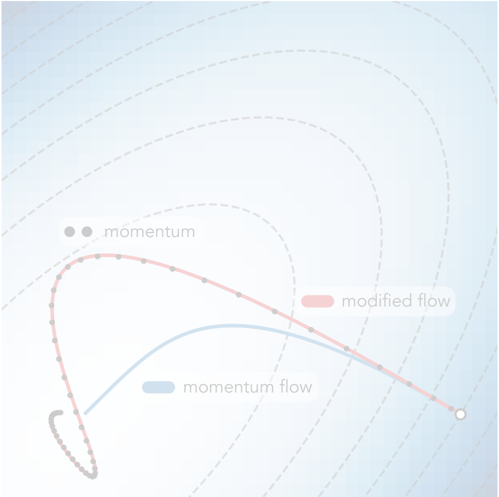
Q. Can we solve for complex learning dynamics of real deep learning models?



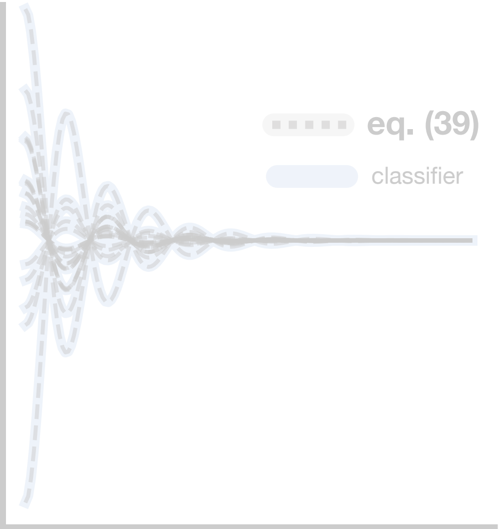
Part 1. Symmetry in the Loss Constrain Gradient and Hessian Geometries



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 θ, i.e. if $f(\psi(\theta, \alpha)) = f(\theta)$ for any (θ, α).

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

Gradient

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

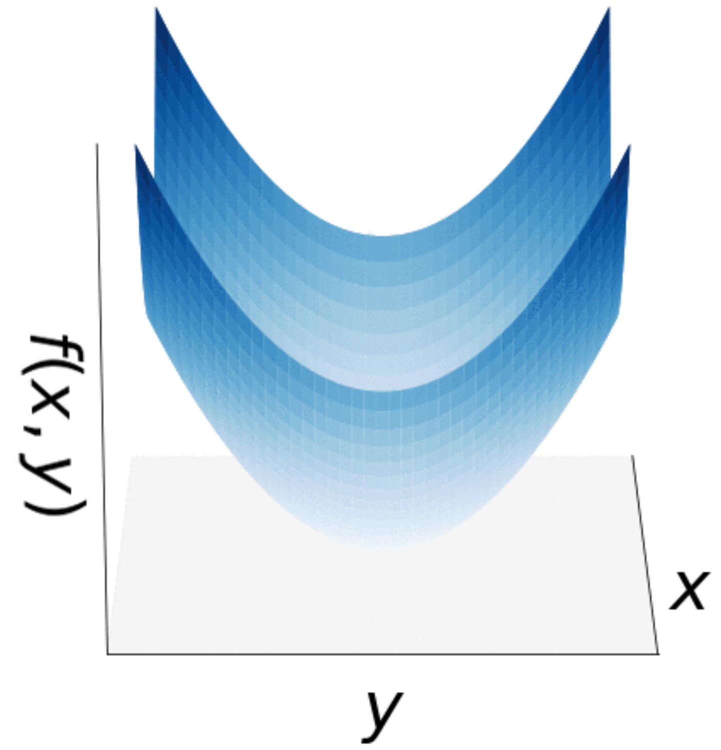
Hessian

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

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

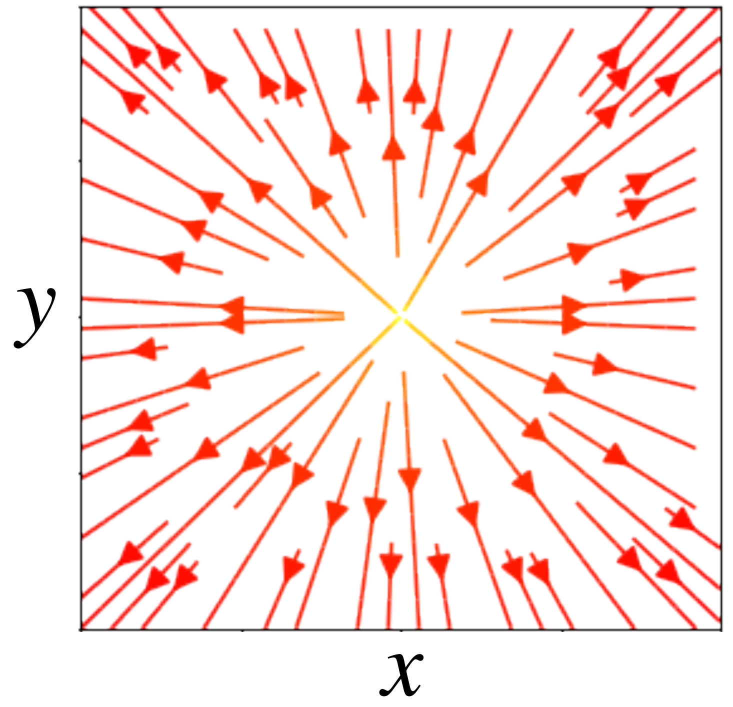
Step 1. Identify symmetry:

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

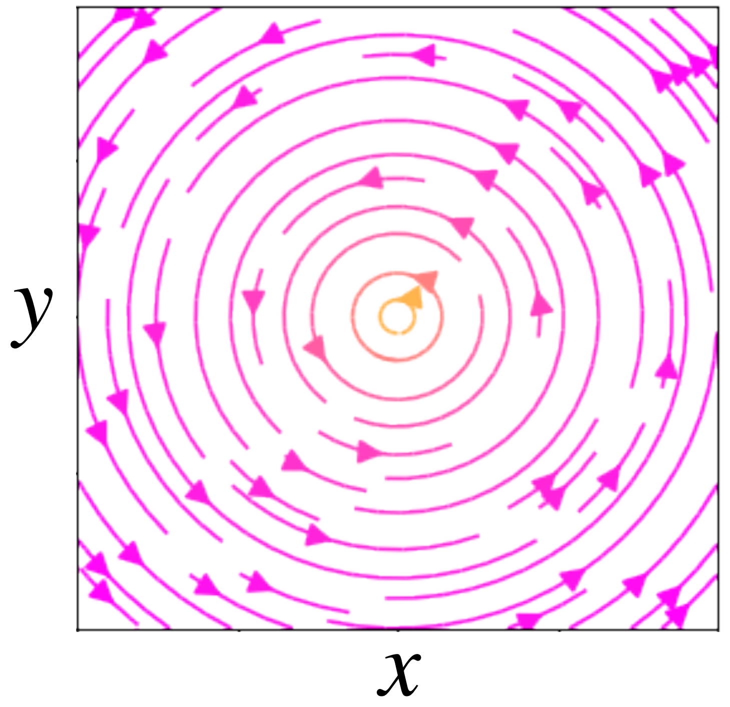


Step 2. Evaluate gradient at identity:

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



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



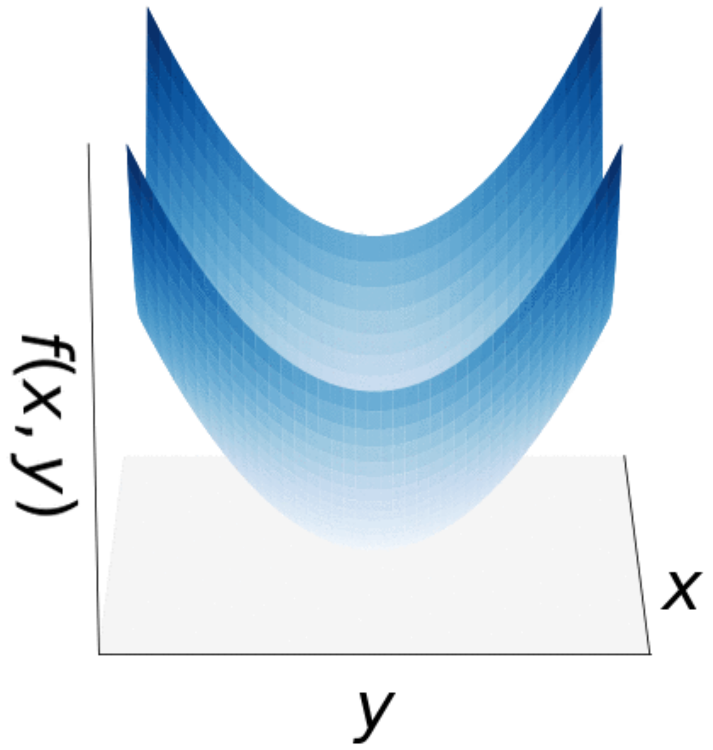
Symmetry Constrain Gradient and Hessian Geometries

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 θ, i.e. if $f(\psi(\theta, \alpha)) = f(\theta)$ for any (θ, α).

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

Gradient	Hessian
$\partial_\alpha f(\psi) = \langle \nabla f, \partial_\alpha \psi \rangle = 0$	$\partial_\theta \partial_\alpha f(\psi) = \mathbf{H} f \partial_\theta \psi \partial_\alpha \psi + \partial_\theta \partial_\alpha \psi \nabla f = 0$

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

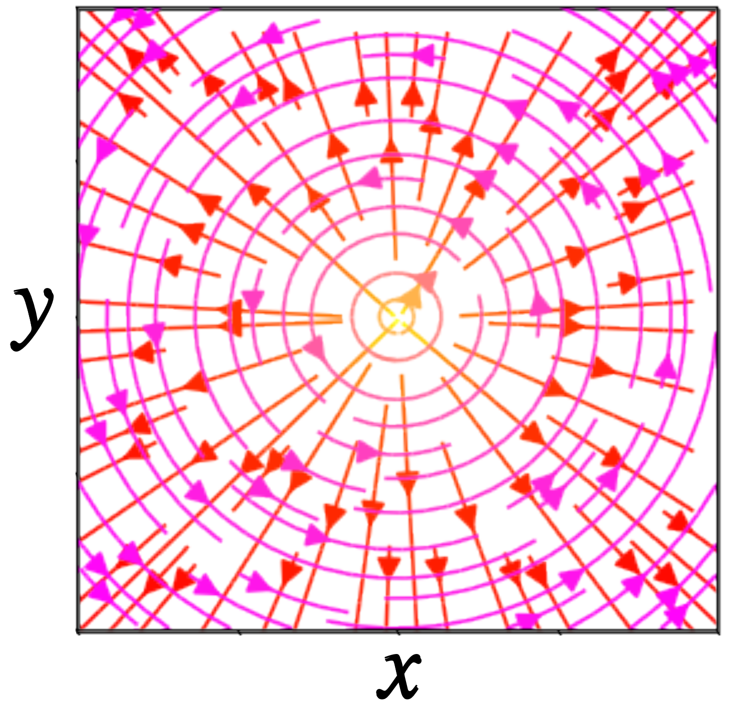


Step 1. Identify symmetry:

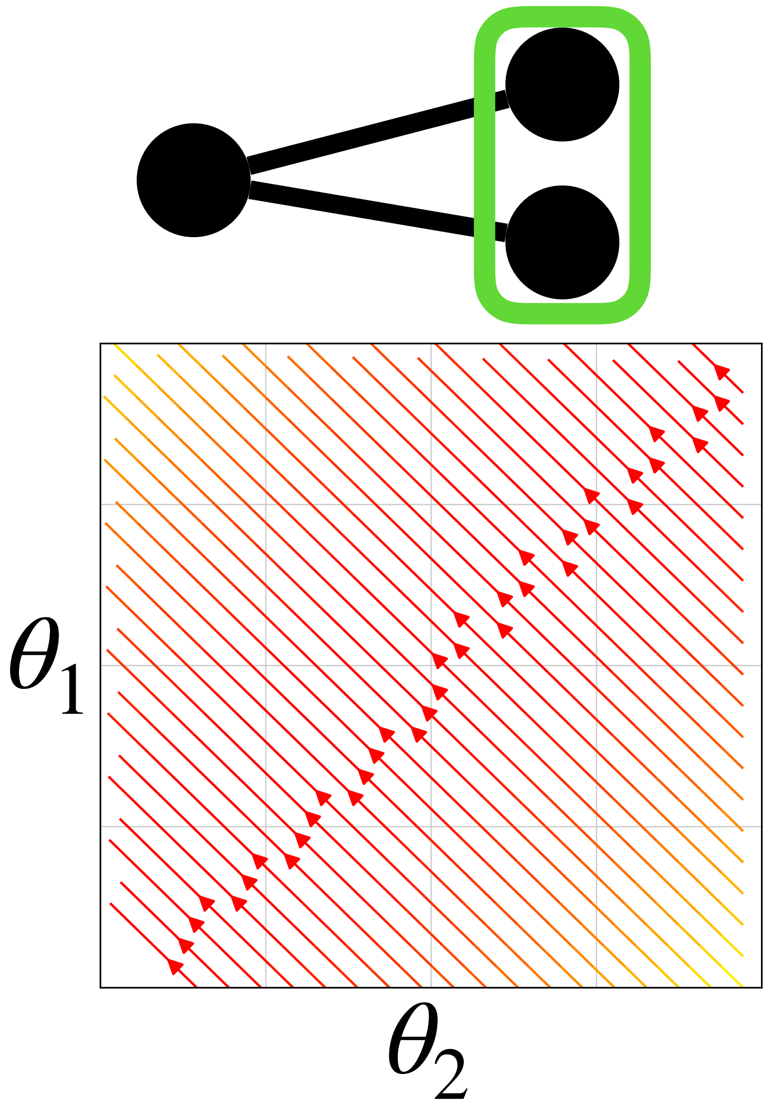
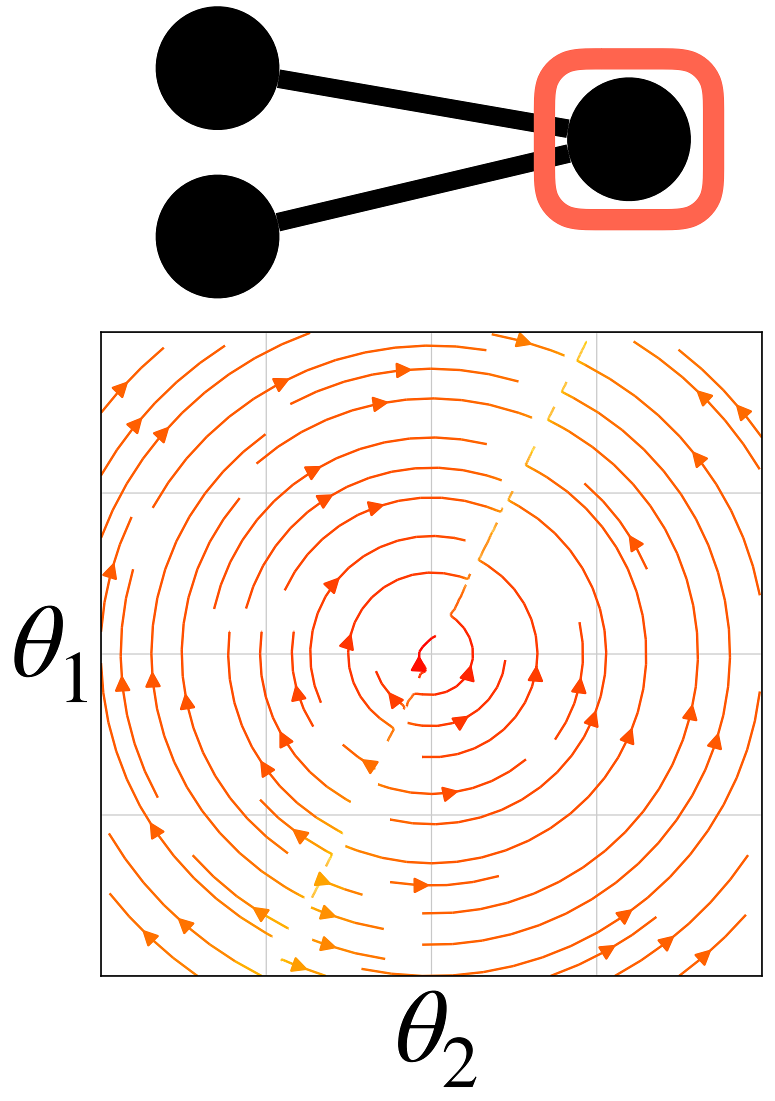
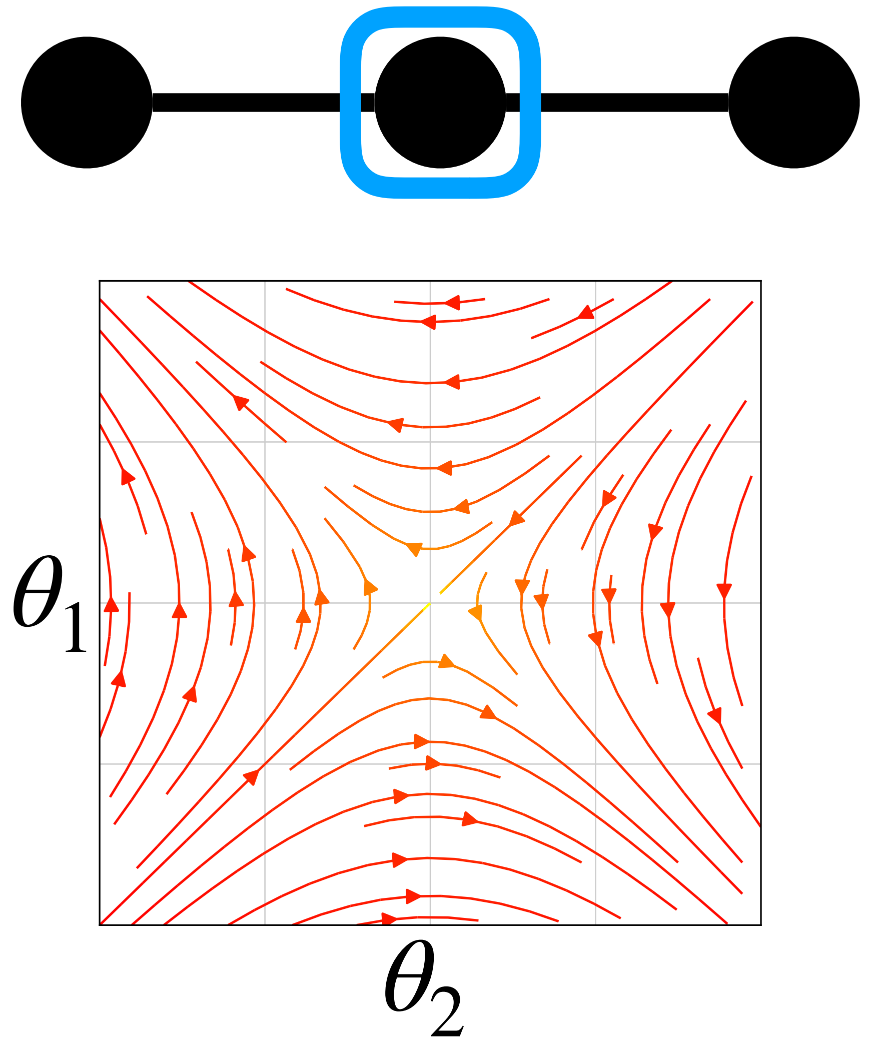
Rotation: $\psi(x, y, \alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$

Step 2. Evaluate gradient at identity:

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



Symmetry resides in all over the modern deep network architectures

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

Symmetry unifies existing literature and yields 15 distinct geometric formulae

Geometric properties of the gradient.

	Translation	Scale	Rescale
$g(\theta) =$	$g\left(\psi(\theta,\alpha)\right)$	$\operatorname{diag}(\alpha_{\mathcal{A}})g\left(\psi(\theta,\alpha)\right)$	$\operatorname{diag}(\alpha_{\mathcal{A}_1} \odot \alpha_{\mathcal{A}_2}^{-1})g\left(\psi(\theta,\alpha)\right)$
$g(\theta) \perp$	$\mathbb{1}_{\mathcal{A}}$	$\theta_{\mathcal{A}}$	$\theta_{\mathcal{A}_1} - \theta_{\mathcal{A}_2}$

Geometric properties of the Hessian.

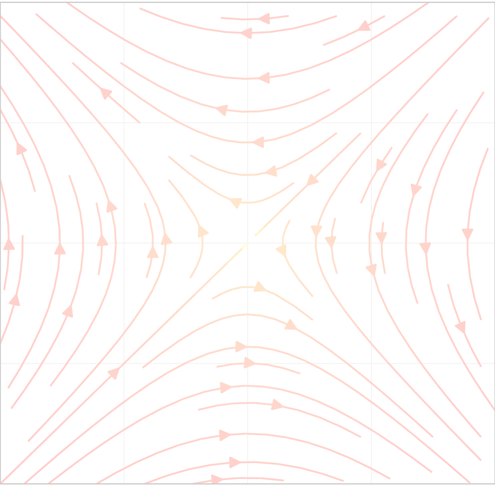
	Translation	Scale	Rescale
$H(\theta) =$	$H(\psi(\theta,\alpha))$	$\operatorname{diag}(\alpha_{\mathcal{A}}^2)H(\psi(\theta,\alpha))$	$\operatorname{diag}(\alpha_{\mathcal{A}_1}^2 \odot \alpha_{\mathcal{A}_2}^{-2})H(\psi(\theta,\alpha))$
$0 =$	$H\mathbb{1}_{\mathcal{A}}$	$H\theta_{\mathcal{A}} + g_{\mathcal{A}}$	$H(\theta_{\mathcal{A}_1} - \theta_{\mathcal{A}_2}) + g_{\mathcal{A}_1} - g_{\mathcal{A}_2}$
$0 =$	$\mathbb{1}_{\mathcal{A}}^{\intercal}H\mathbb{1}_{\mathcal{A}}$	$\theta_{\mathcal{A}}^{\intercal}H\theta_{\mathcal{A}}$	$(\theta_{\mathcal{A}_1} - \theta_{\mathcal{A}_2})^{\intercal}H(\theta_{\mathcal{A}_1} - \theta_{\mathcal{A}_2}) + g_{\mathcal{A}_1}\theta_{\mathcal{A}_1} + g_{\mathcal{A}_2}\theta_{\mathcal{A}_2}$

Unifying existing literature through symmetry.

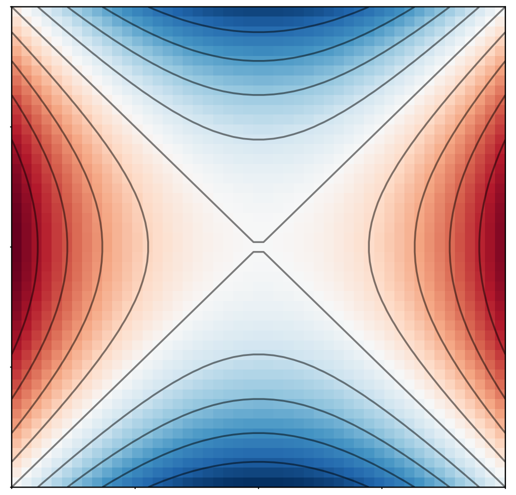
		Translation	Scale	Rescale
∇F	$\partial_\theta F$	—	1, 2, 5	6
	$\partial_\alpha F$	—	3, 4	7,8,9
$\mathbf{H}F$	$\partial_\theta^2 F$	—	3, 5	—
	$\partial_\theta\partial_\alpha F$	—	—	—
	$\partial_\alpha^2 F$	—	—	9

1. *Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. 2015*
2. *Twan Van Laarhoven. L2 regularization versus batch and weight normalization. 2017*
3. *Guodong Zhang, Chaoqi Wang, Bowen Xu, and Roger Grosse. Three mechanisms of weight decay regularization. 2018*
4. *Zhiyuan Li, Kaifeng Lyu, and Sanjeev Arora. Reconciling modern deep learning with traditional optimization analyses: The intrinsic learning rate. 2020*
5. *Behnam Neyshabur, Russ R Salakhutdinov, and Nati Srebro. Path-sgd: Path-normalized optimization in deep neural networks. 2015*
6. *Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. 2018*
7. *Simon S Du, Wei Hu, and Jason D Lee. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. 2018*
8. *Tengyuan Liang, Tomaso Poggio, Alexander Rakhlin, and James Stokes. Fisher-rao metric, geometry, and complexity of neural networks. 2019*
9. *Hidenori Tanaka*, Daniel Kunin*, Daniel LK Yamins, and Surya Ganguli. Pruning neural networks without any data by iteratively conserving synaptic flow. 2020.*

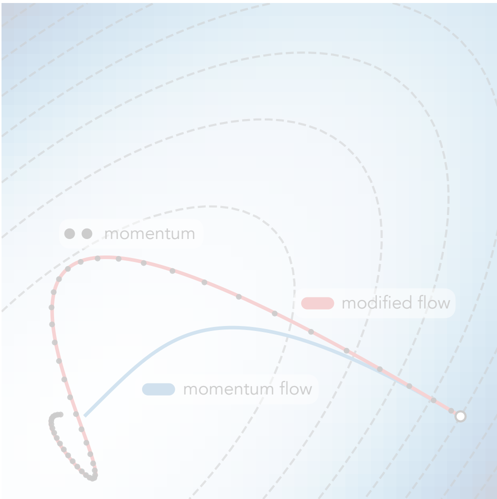
Q. Can we solve for complex learning dynamics of real deep learning models?



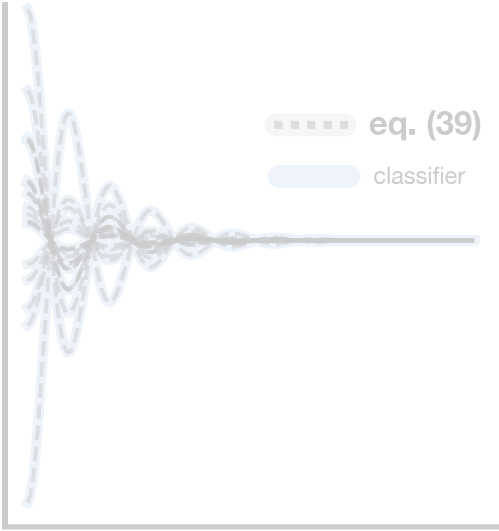
Part 1. Symmetry in the Loss Constrain Gradient and Hessian Geometries



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

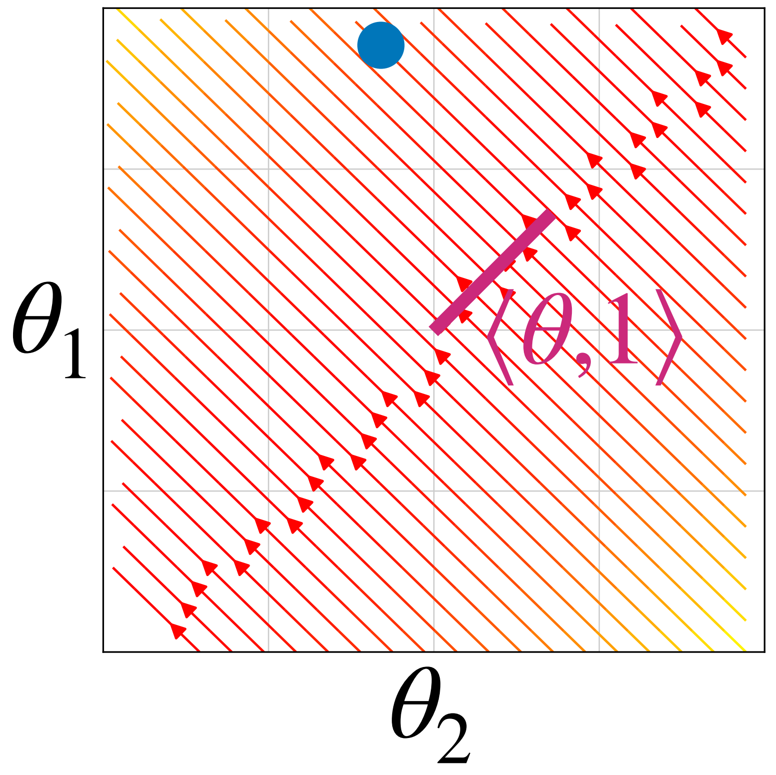
Symmetry Leads to Conservation Laws Under Gradient Flow

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

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

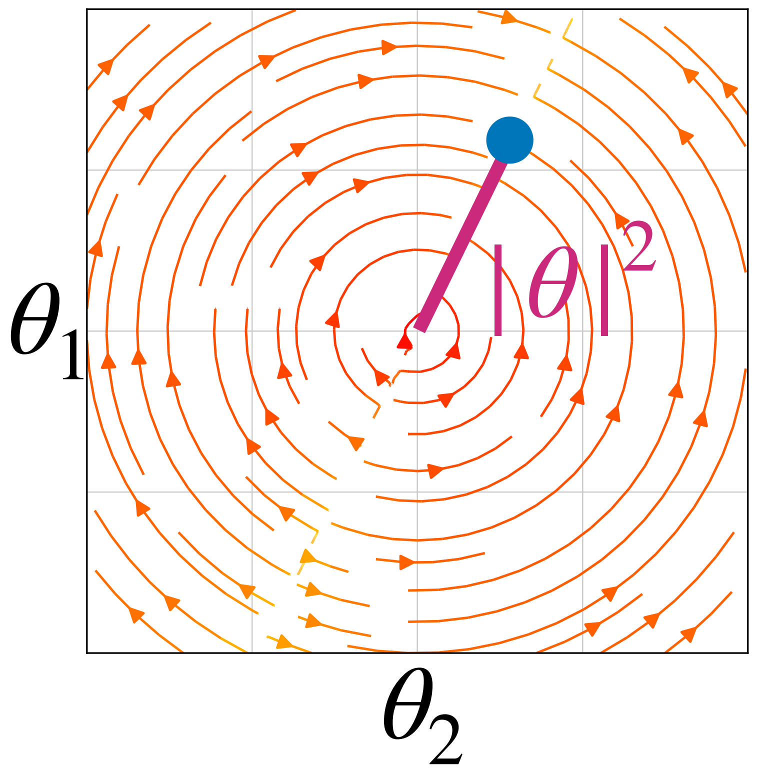
How do these learning dynamics interact with the geometric properties introduced by symmetry?

Translation



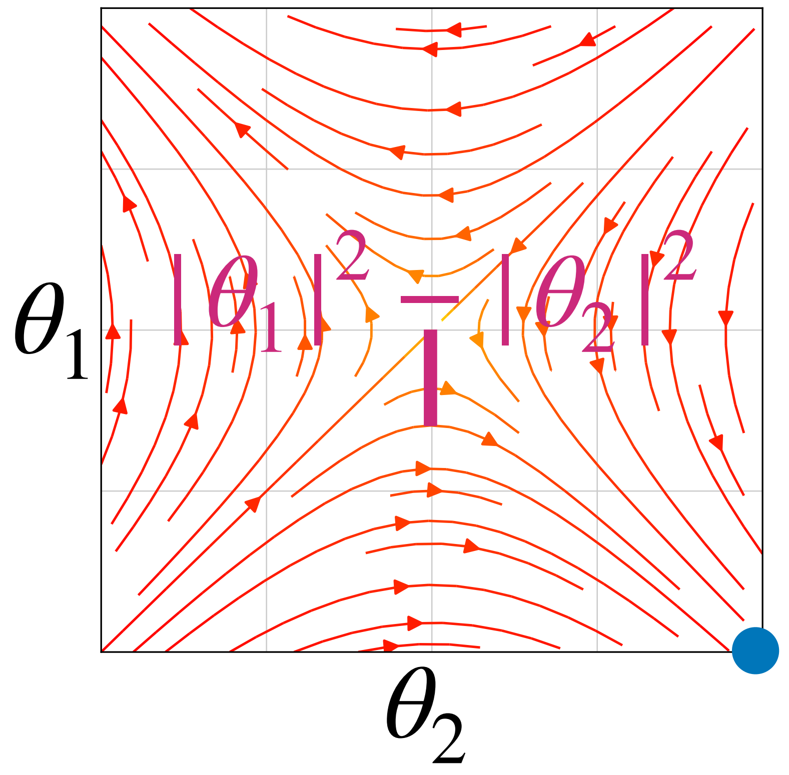
$$\langle\theta_A(t), 1\rangle = \langle\theta_A(0), 1\rangle$$

Scale



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

Rescale



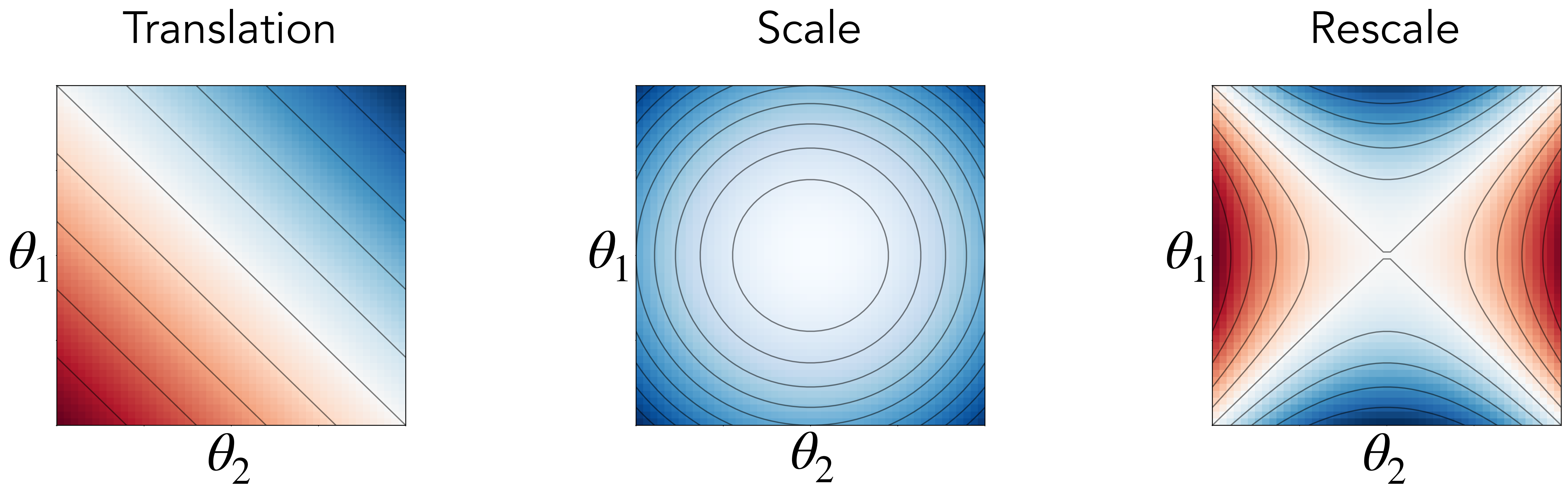
$$|\theta_{A_1}(t)|^2 - |\theta_{A_2}(t)|^2 = |\theta_{A_1}(0)|^2 - |\theta_{A_2}(0)|^2$$

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 η 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?



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

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

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

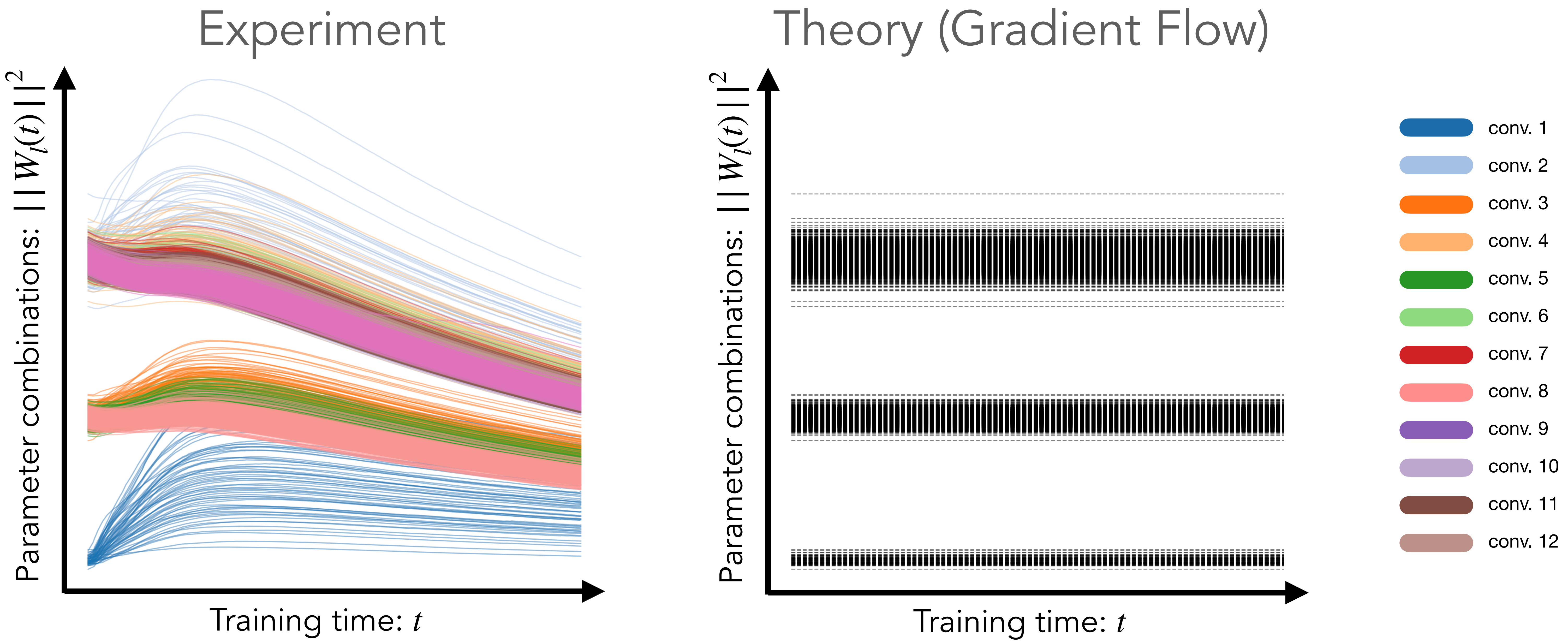
A version of Noether's Theorem: Every symmetry* of a network architecture has a corresponding conserved quantity through training under gradient flow. Projecting the gradient flow dynamics onto the generator vector field generates an ODE, whose solution is a conservation law.

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

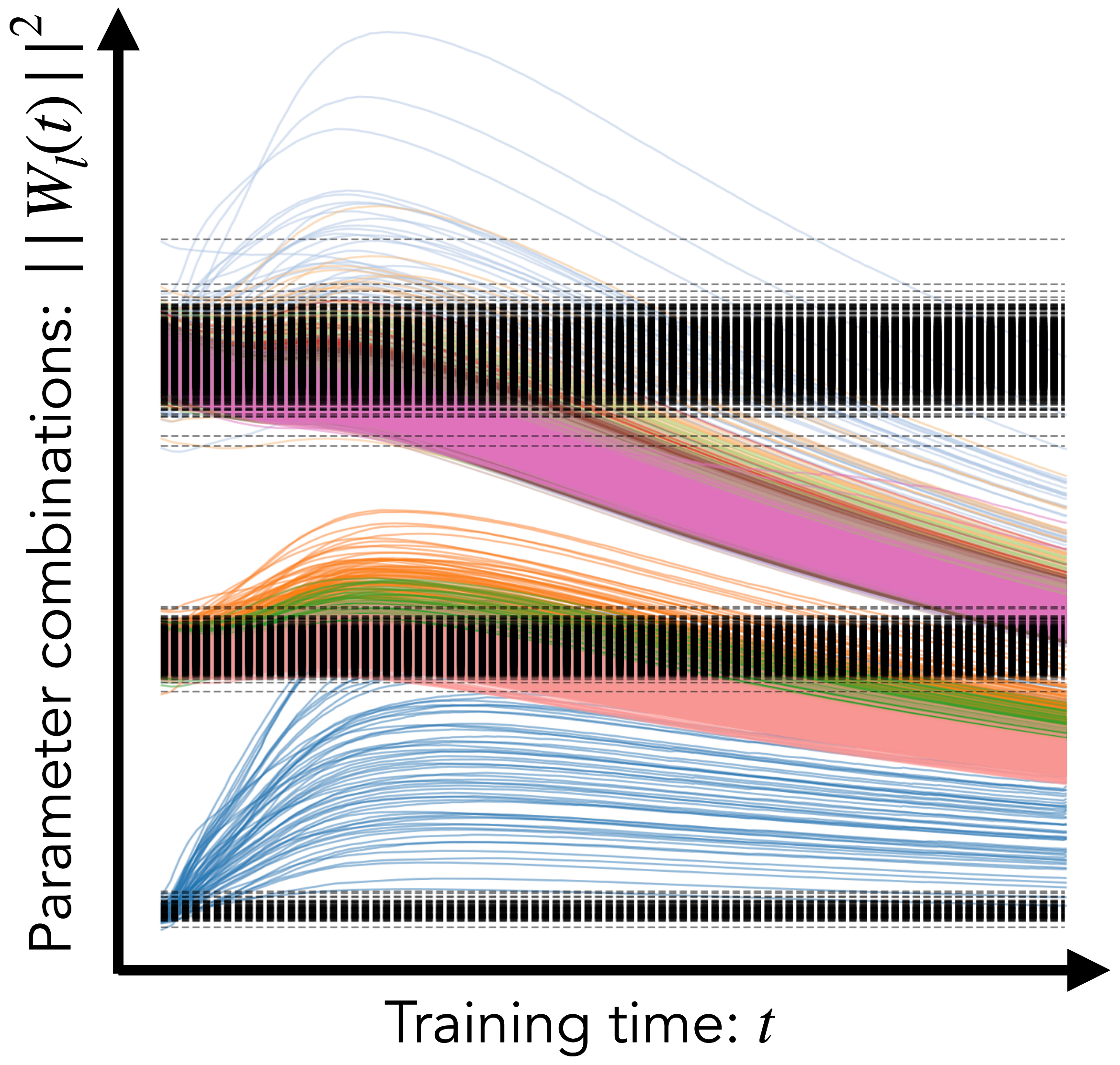
*satisfying a mild assumption

Emmy Noether (1882 - 1935)

Does this theory agree with empirics?



No, conservation laws are broken empirically!

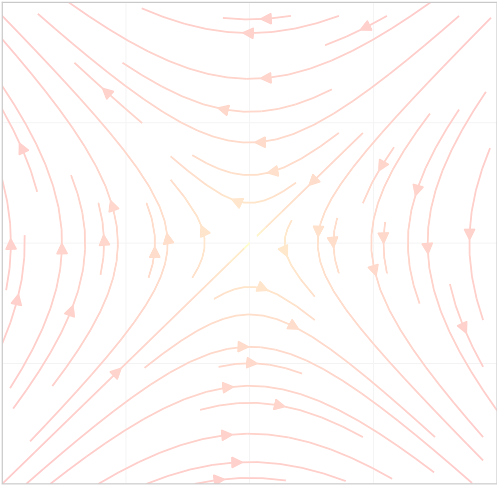


Q. Why?

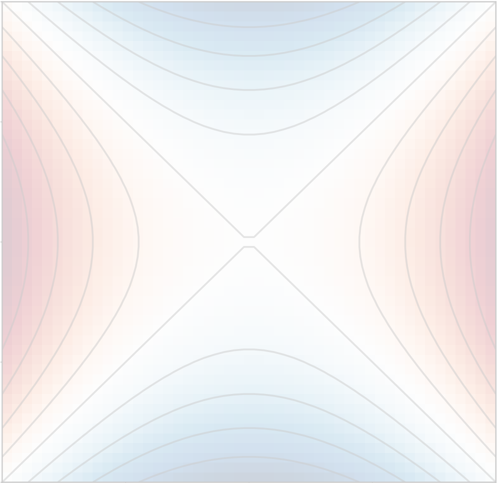
Gradient flow is too simple of a continuous model for SGD.
It fails to account for key building blocks of modern optimization:

- weight decay
- momentum
- stochasticity
- finite learning rates

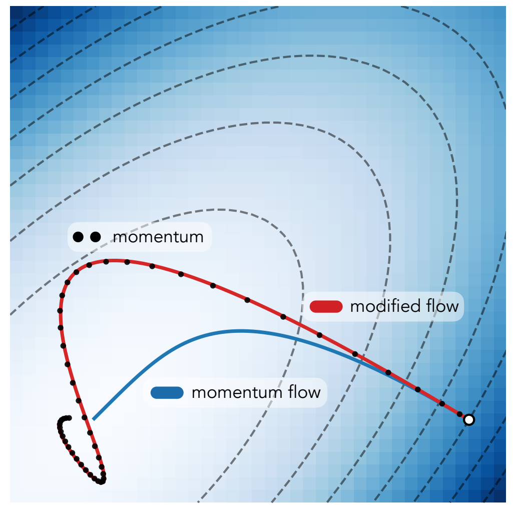
Q. Can we solve for complex learning dynamics of real deep learning models?



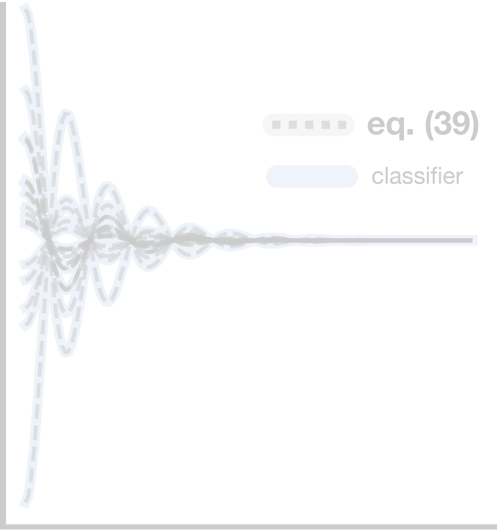
Part 1. Symmetry in the Loss Constrain Gradient and Hessian Geometries



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 (λ): Weight decay changes the trajectory from gradient flow pulling the network to the origin in parameter space.

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

Modeling momentum (β): Momentum accelerates the learning dynamics rescaling time, but leaves the trajectory intact.

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

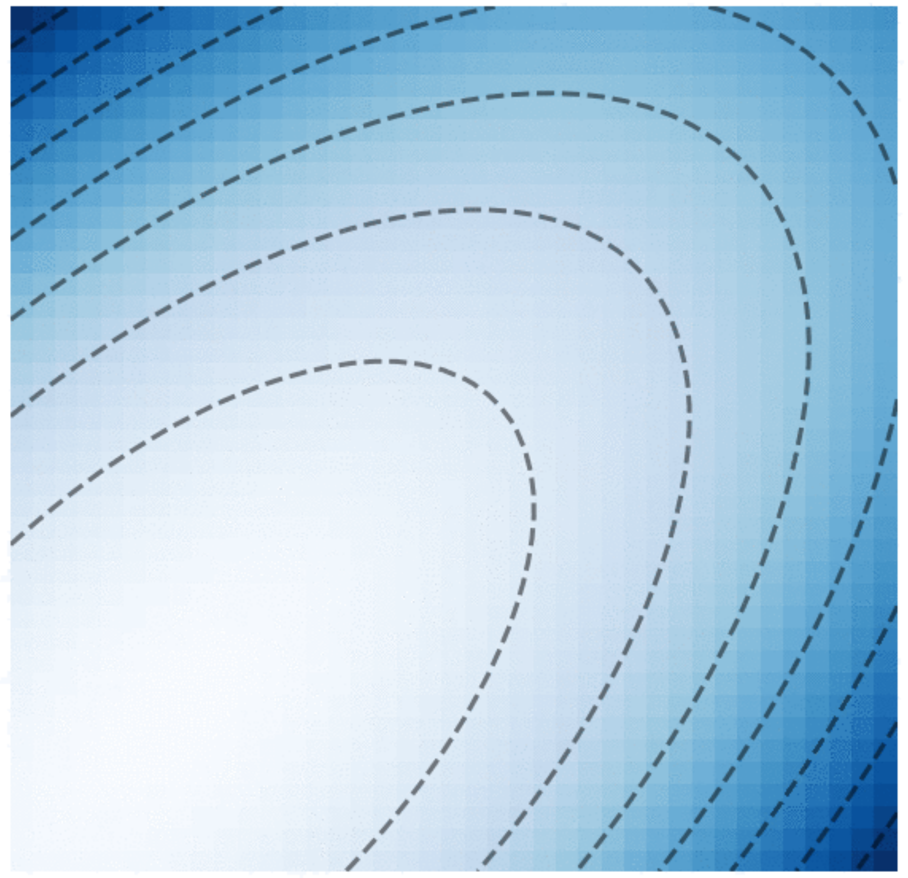
Modeling stochasticity: We model the batch gradient $\hat{g}_{\mathscr{B}}(\theta)$ as a noisy version of the full batch gradient $g(\theta)$ such that,

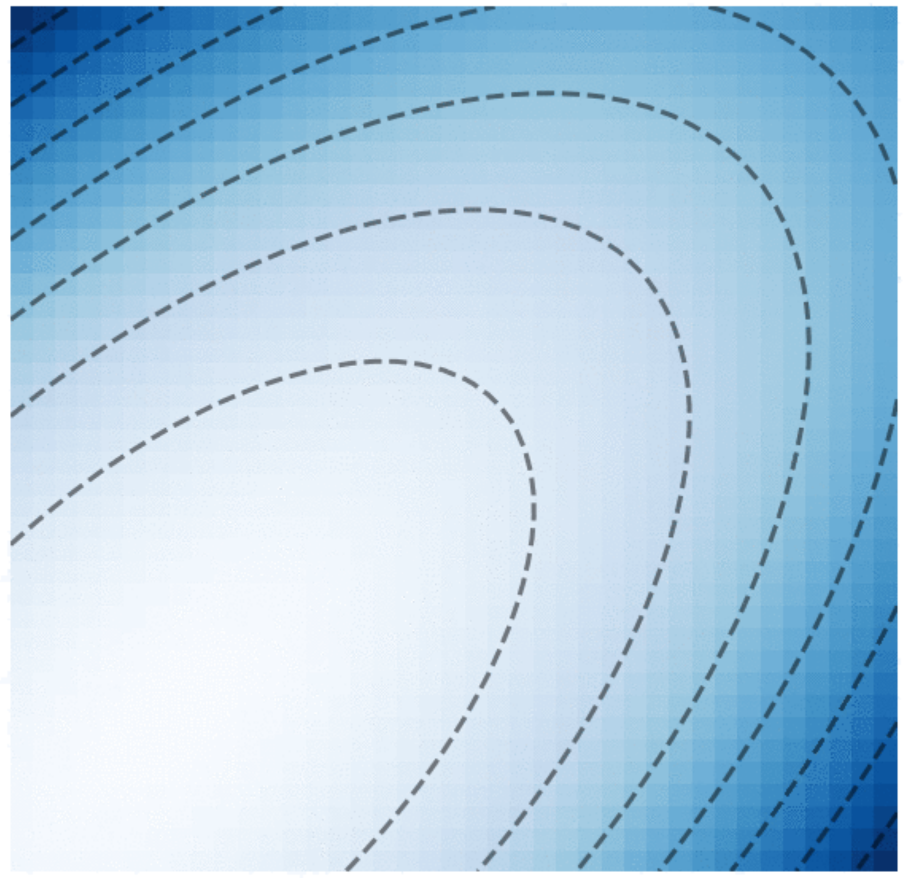
$$\hat{g}_{\mathscr{B}}(\theta) = g(\theta) + \epsilon$$

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

Example: Quadratic Loss

$$\mathscr{L} = \theta^\intercal A \theta$$





Blue curve: gradient flow

Red curve: modified trajectory

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.

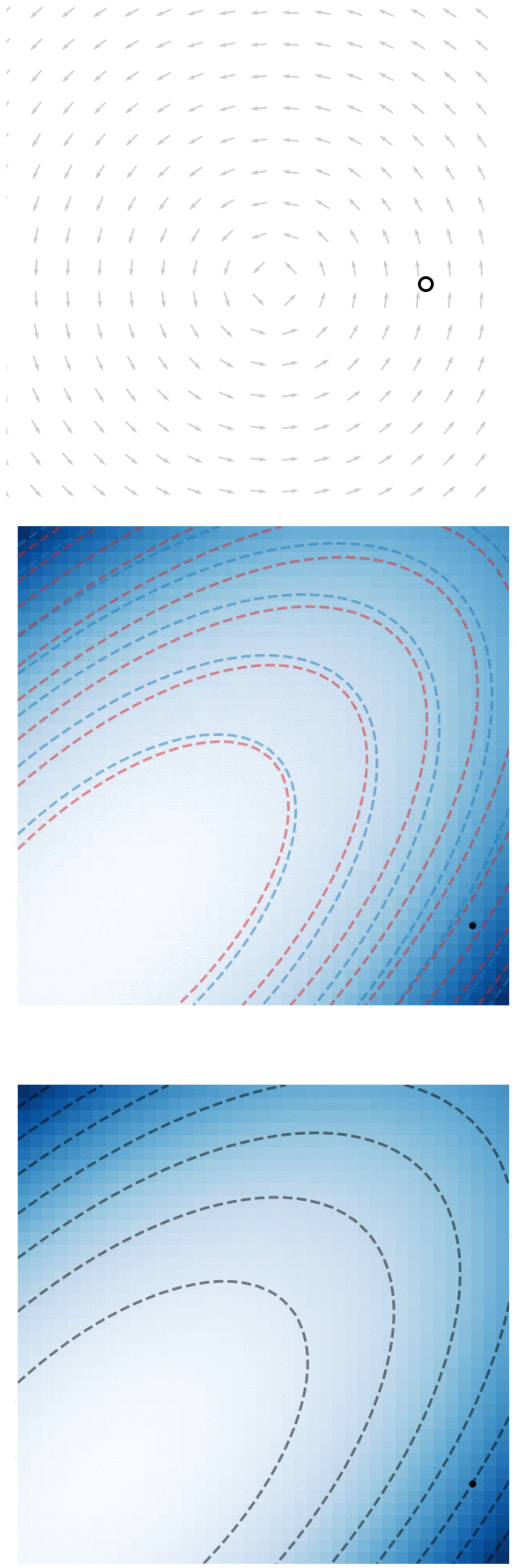
$$\frac{d\theta}{dt} = -g(\theta) - \frac{\eta}{2}H(\theta)g(\theta)$$

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

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

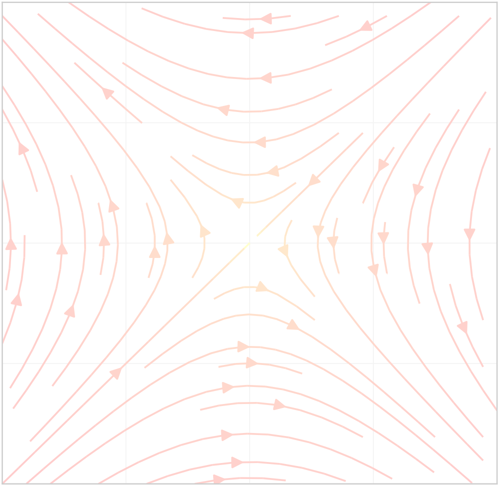
$$\frac{d\theta}{dt} = -g(\theta) - \frac{\eta}{2}\frac{d^2\theta}{dt^2}$$

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

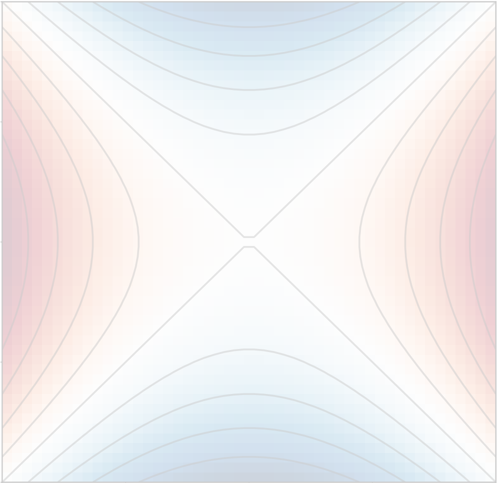


Blue curve: gradient flow
Red curve: modified trajectory
Black dots: discrete SGD steps

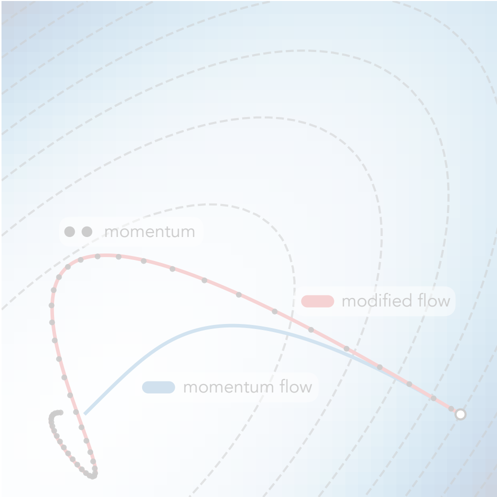
Q. Can we solve for complex learning dynamics of real deep learning models?



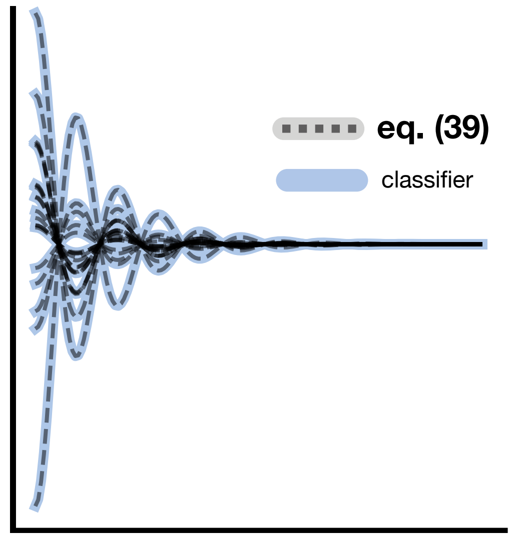
Part 1. Symmetry in the Loss Constrain Gradient and Hessian Geometries



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?

1. Consider a realistic continuous model for SGD (weight decay and modified loss).

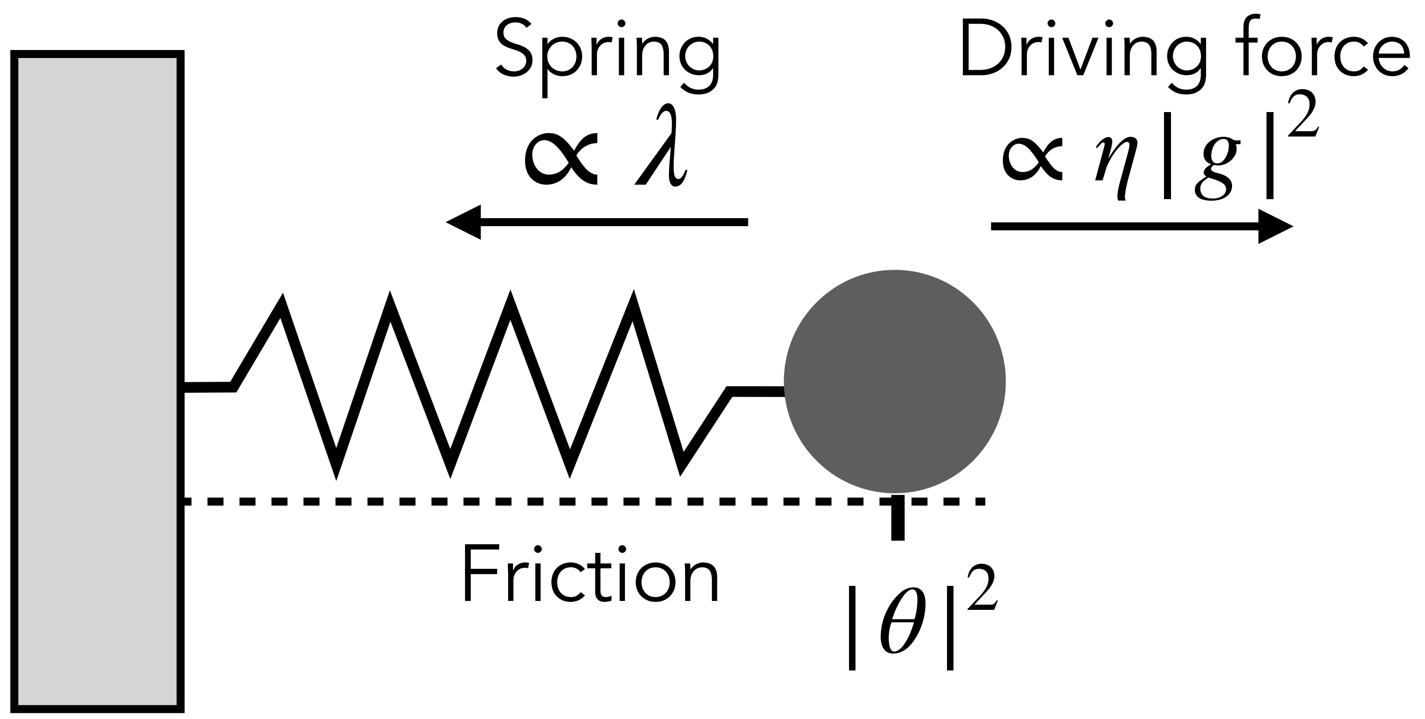
$$\frac{d\theta}{dt} + \lambda\theta = -g - \frac{\eta}{2}Hg$$

2. Project the learning dynamics onto the generator vector fields. (e.g. $\partial_\alpha \psi = \partial_\alpha(\alpha\theta) = \theta$).
3. Harness the geometric constraints introduced by symmetry.

$$\left\langle \frac{d\theta}{dt}, \theta \right\rangle + \lambda\langle\theta,\theta\rangle = -\overbrace{\cancel{\langle g,\theta\rangle}}^{\langle g,\theta\rangle = 0} - \frac{\eta}{2}\overbrace{\langle Hg,\theta\rangle}^{\langle Hg,\theta\rangle = -|g|^2}$$

4. Solve the resulting ODE.

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



Overdamped driven oscillator

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

VGG-16 trained on Tiny ImageNet with SGD

Translation

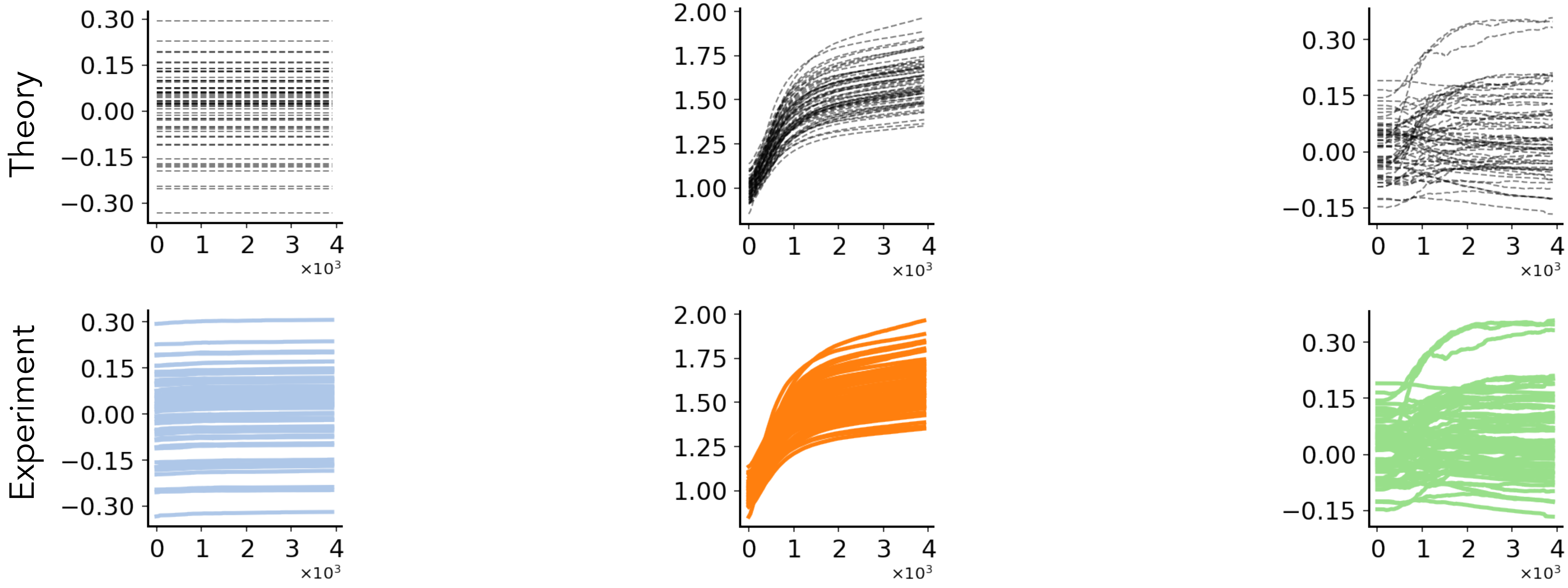
$$\langle \theta_{\mathcal{A}}(t), \mathbb{1}\rangle = e^{-\lambda t}\langle \theta_{\mathcal{A}}(0), \mathbb{1}\rangle$$

Scale

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

Rescale

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

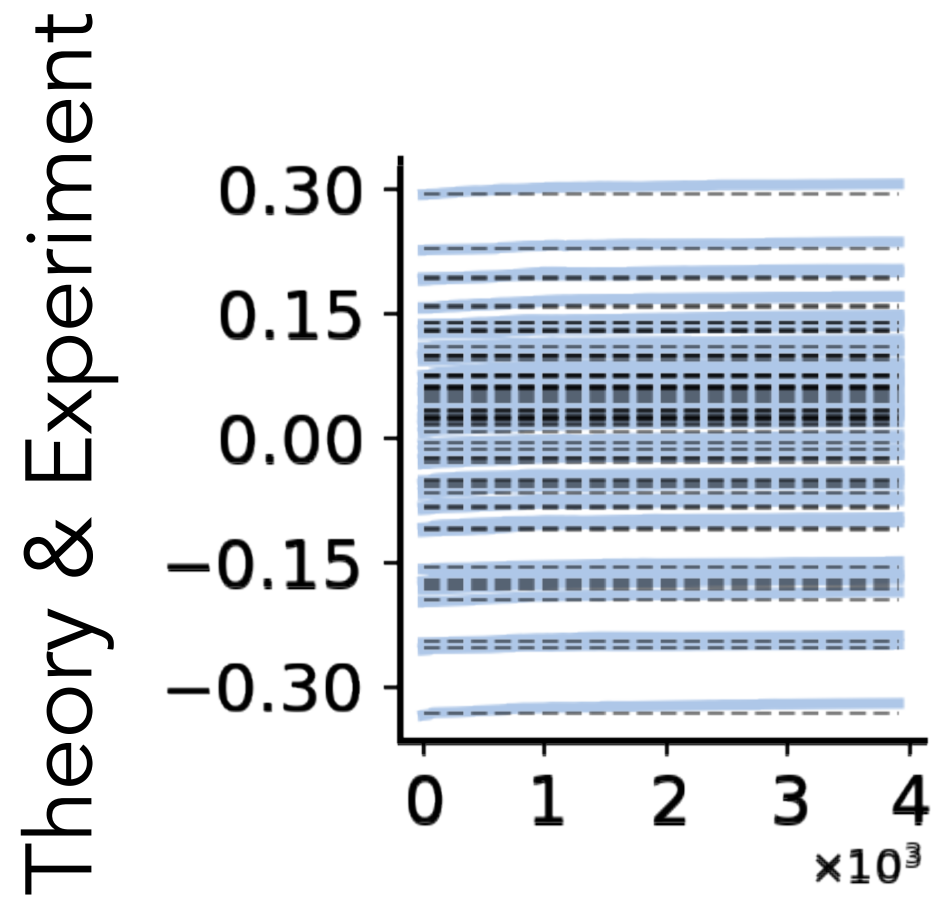


Theory (dotted lines) match the empirics (colored lines) perfectly!

VGG-16 trained on Tiny ImageNet with SGD

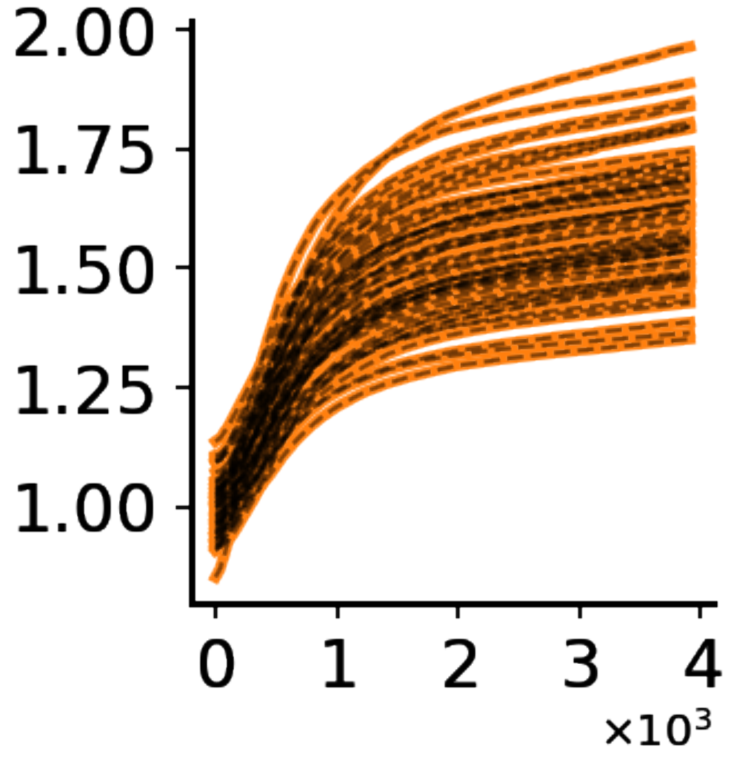
Translation

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



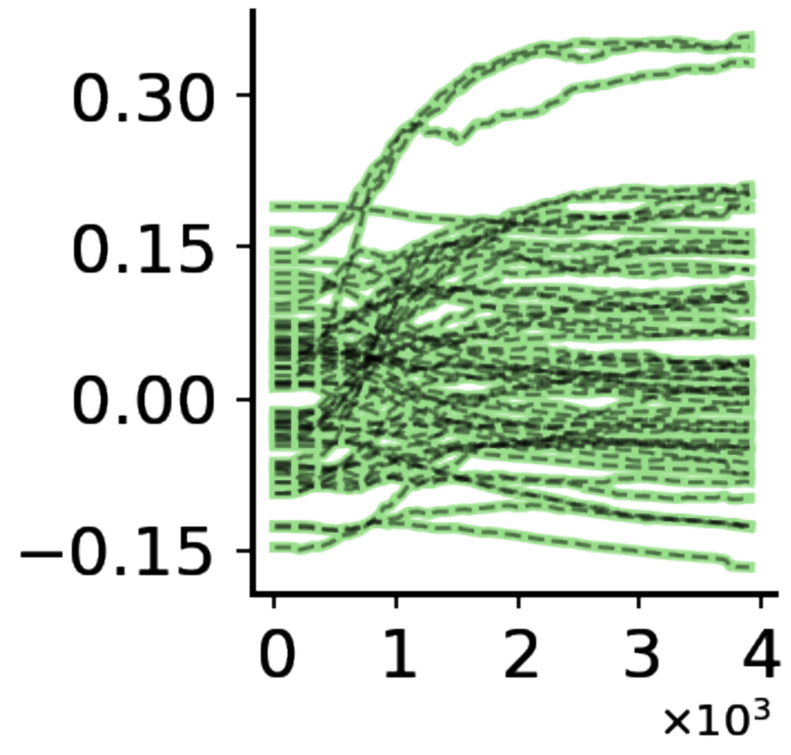
Scale

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



Rescale

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

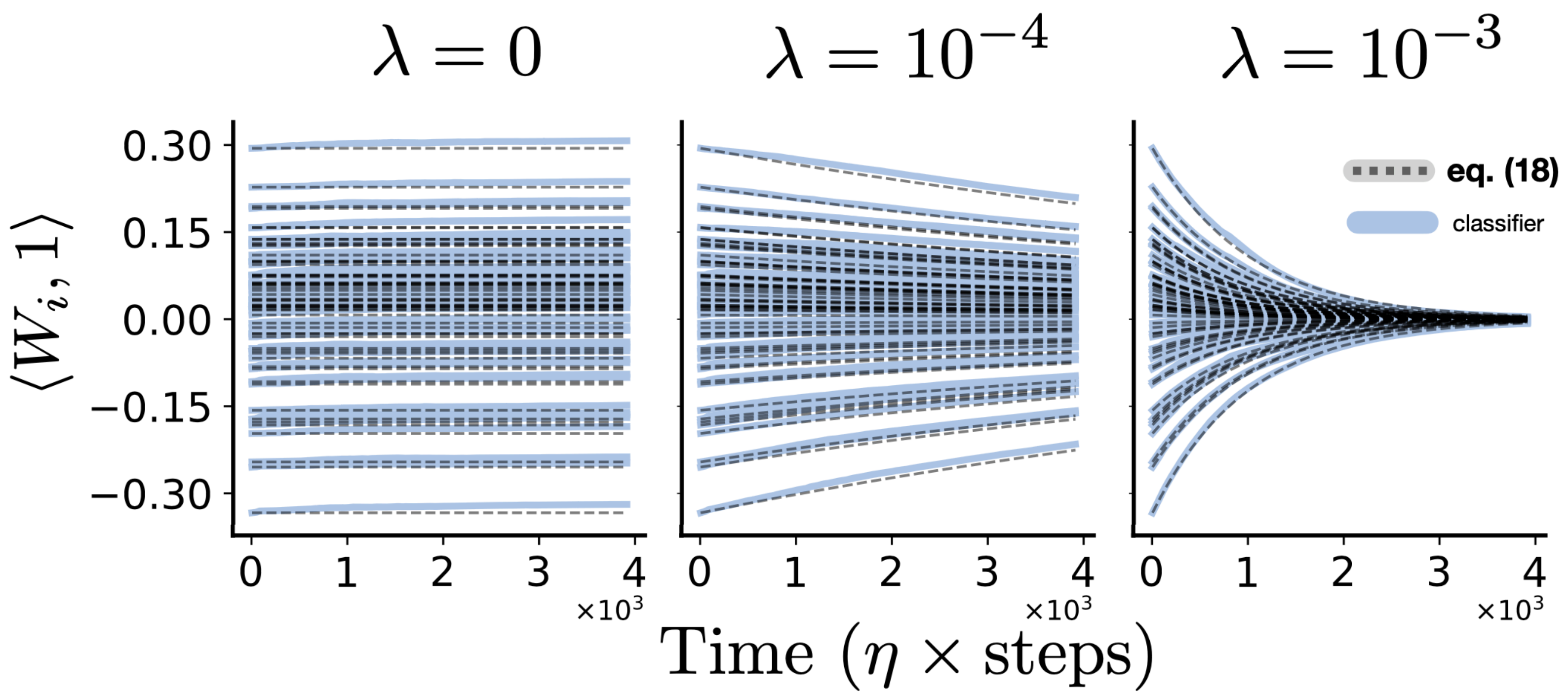


Theory (dotted lines) match the empirics (colored lines) perfectly!

VGG-16 trained on Tiny ImageNet with SGD

Translation

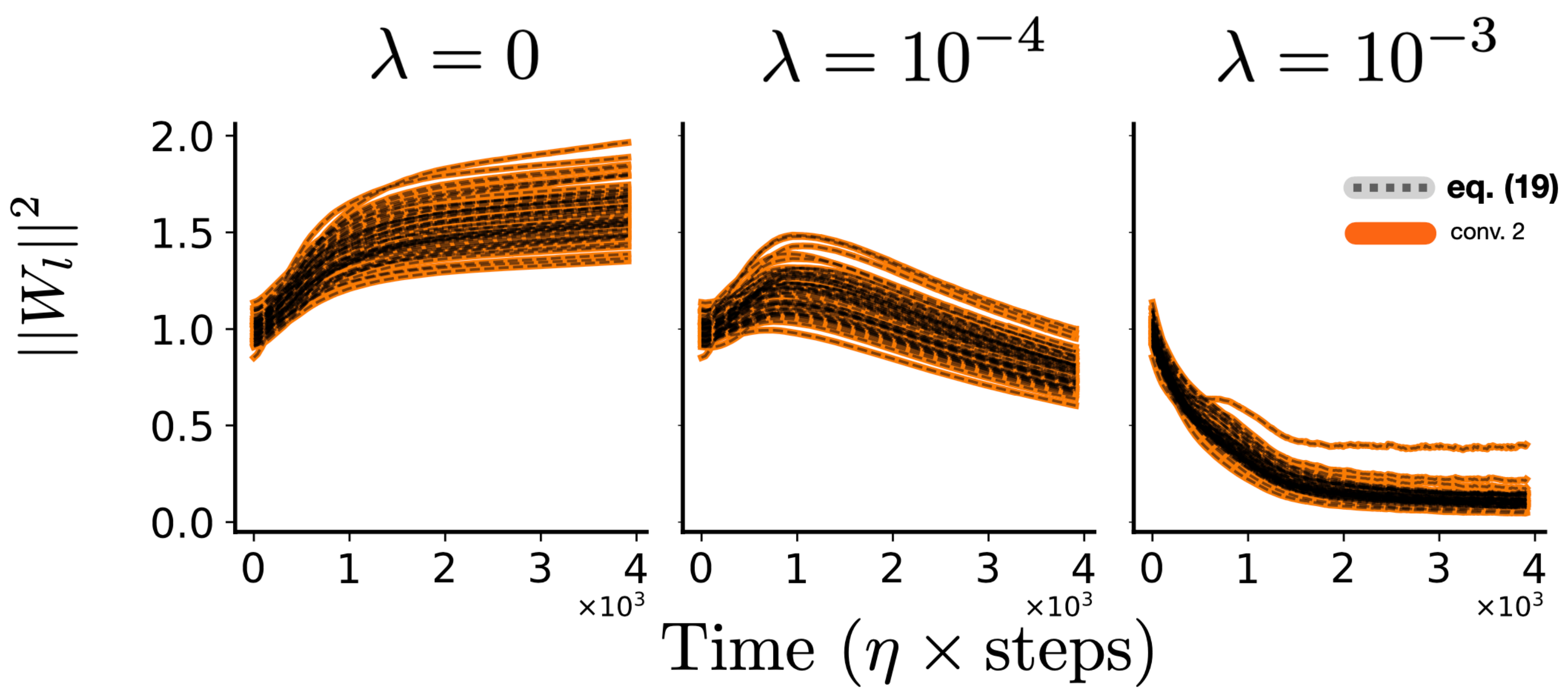
$$\langle \theta_{\mathcal{A}}(t), \mathbb{1}\rangle = e^{-\lambda t}\langle \theta_{\mathcal{A}}(0), \mathbb{1}\rangle$$



- $\langle\theta_{\mathscr{A}}(t),1\rangle$ 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

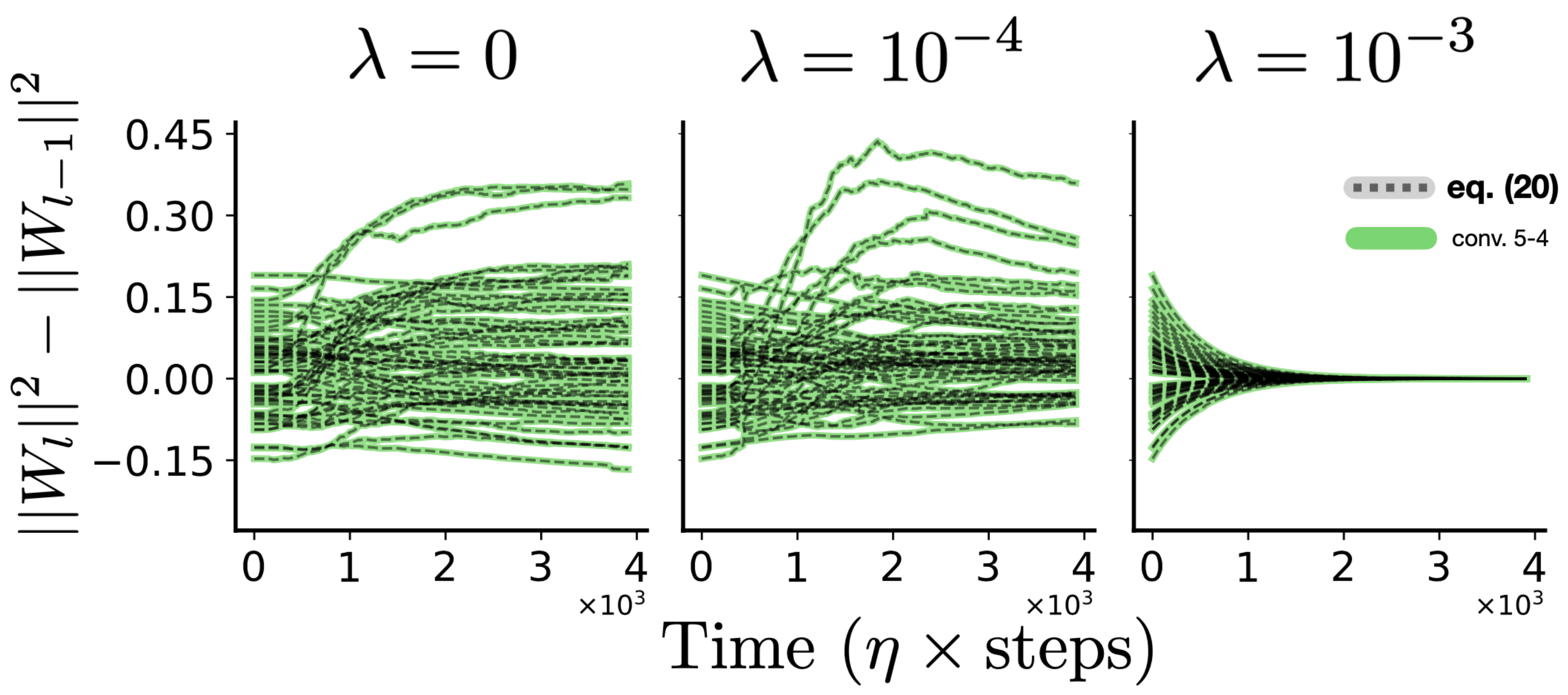
$$|\theta_{\mathcal{A}}(t)|^2 = e^{-2\lambda t}|\theta_{\mathcal{A}}(0)|^2 + \eta \int_0^t e^{-2\lambda(t-\tau)} |g_{\mathcal{A}}|^2 \, d\tau$$



- Norm $|\theta_{\mathscr{A}}|^2$ 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

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



- 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.

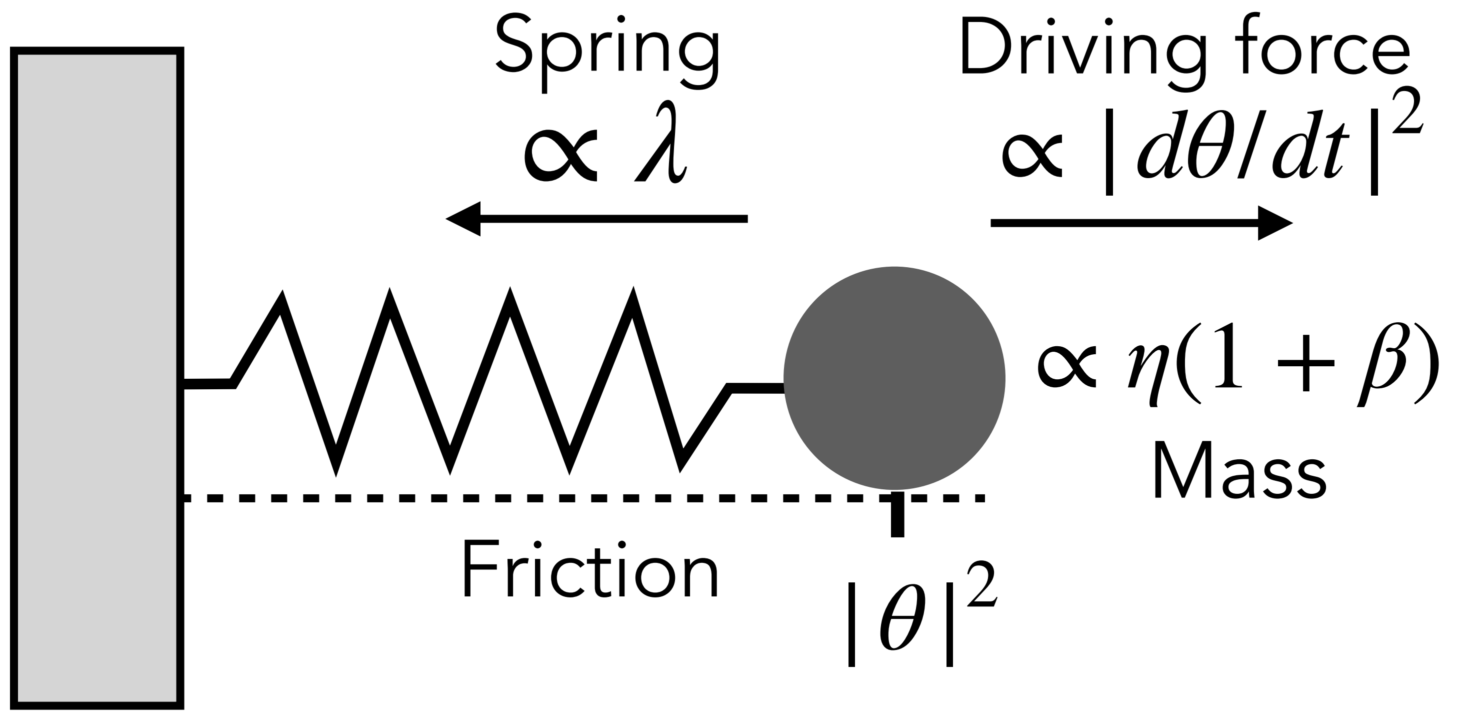
Q. How do weight decay, momentum, stochastic gradients, and finite learning rates all interact to break these conservation laws?

The continuous model for SGD with weight decay, momentum, and modified flow:

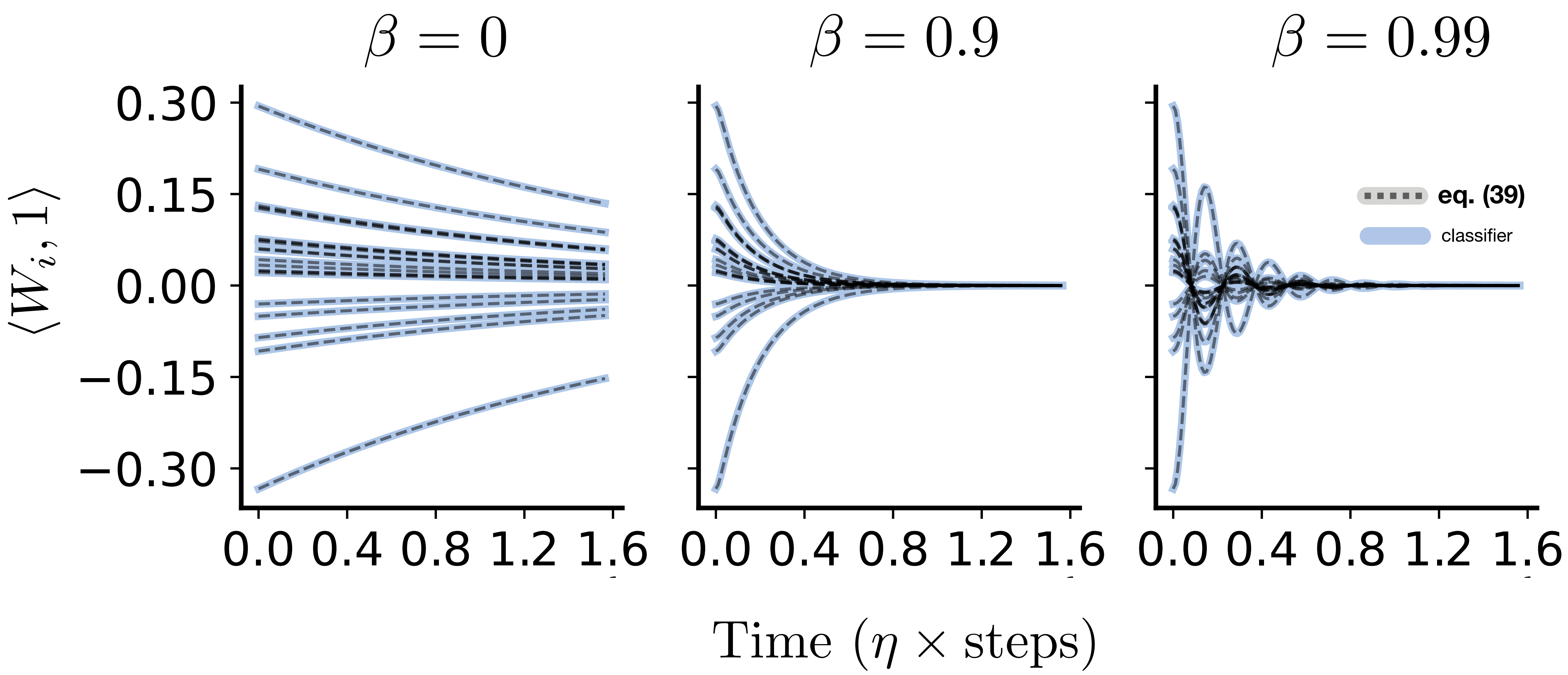
$$\frac{\eta(1+\beta)}{2}\frac{d^2\theta}{dt^2} + (1-\beta)\frac{d\theta}{dt} + \lambda\theta = -g$$

The ODE after projecting the dynamics:

$$\frac{\eta(1+\beta)}{2}\frac{d^2|\theta|^2}{dt^2} + (1-\beta)\frac{d|\theta|^2}{dt} + 2\lambda|\theta|^2 = \eta(1+\beta)\left|\frac{d\theta}{dt}\right|^2$$



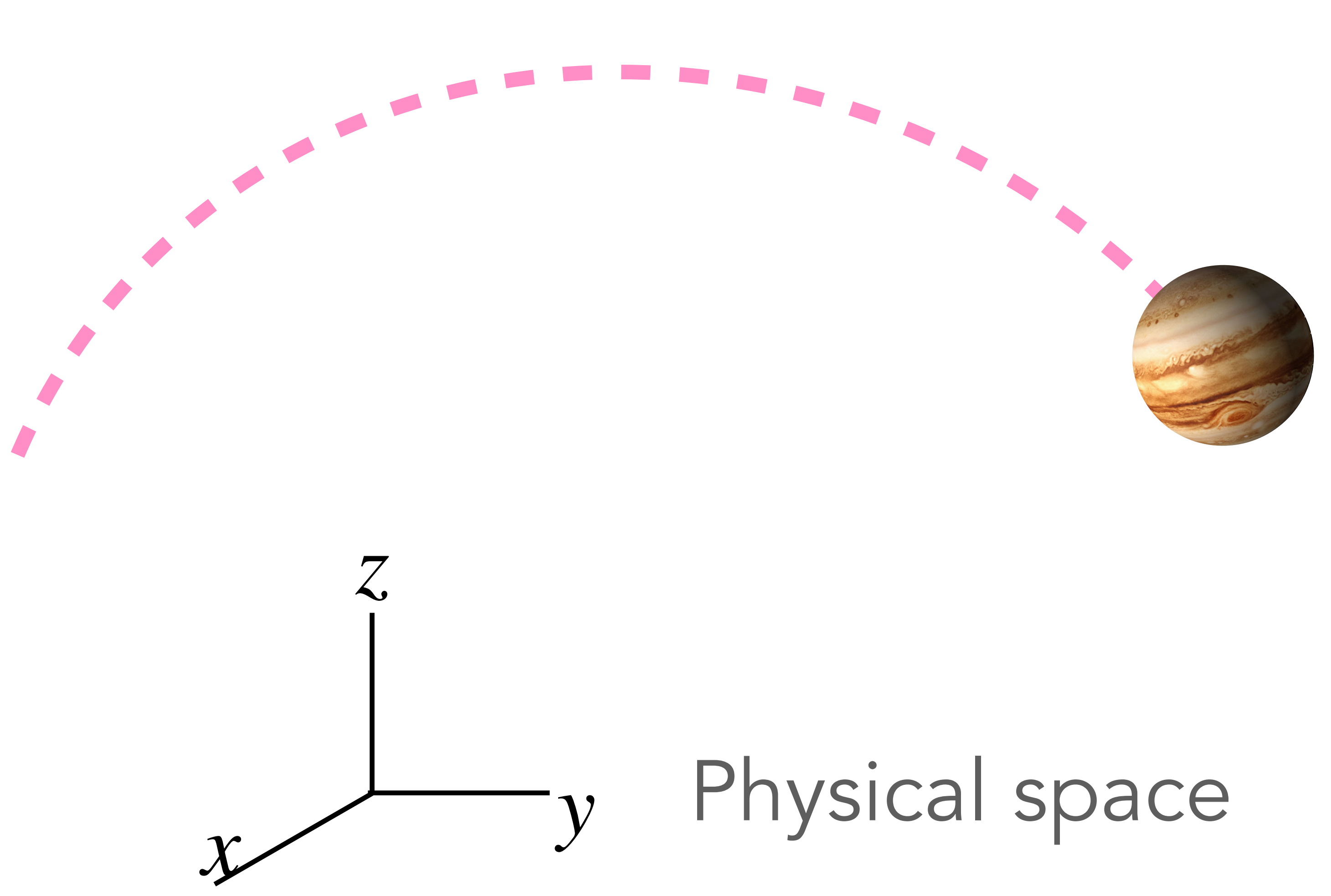
Underdamped driven oscillator

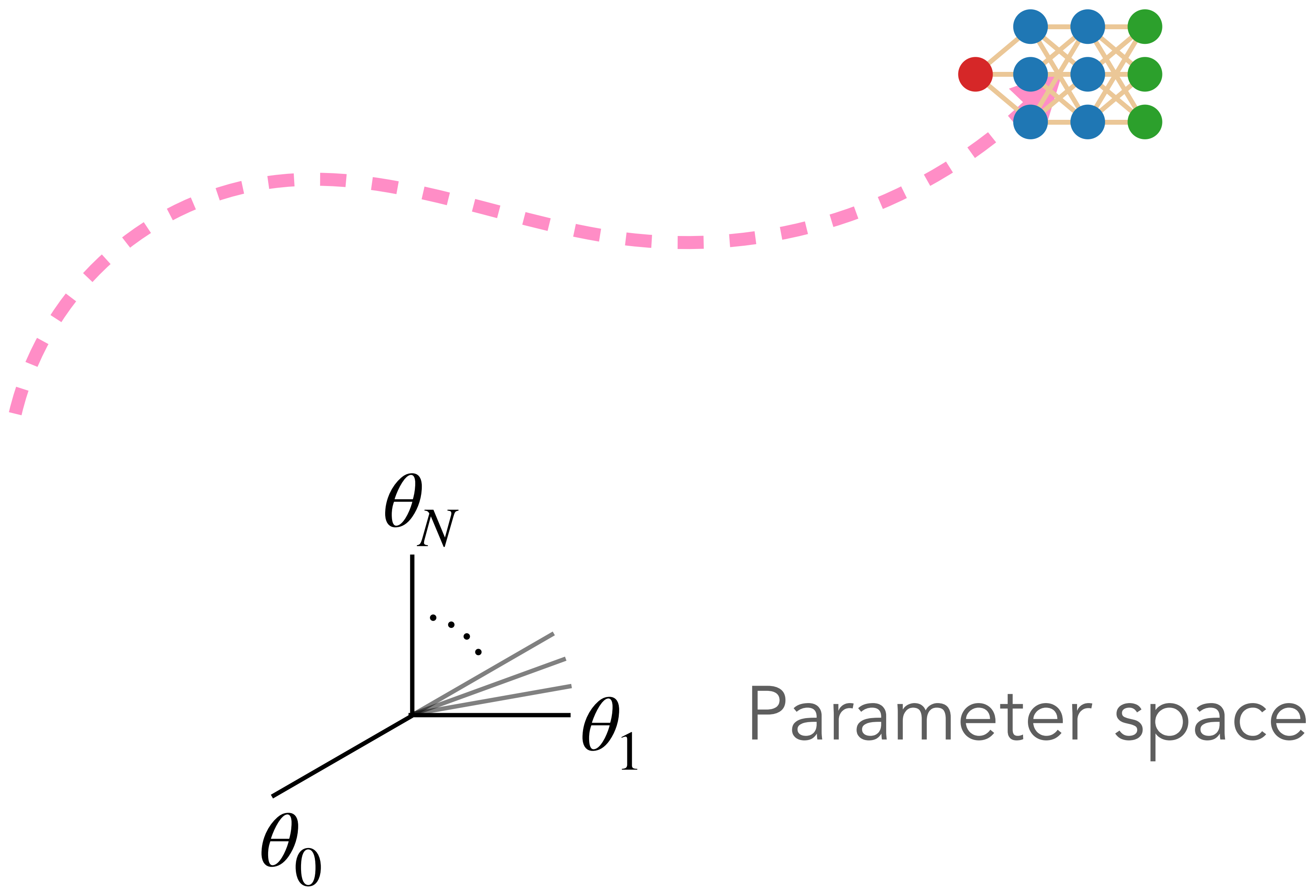


VGG-16 trained on Tiny ImageNet with SGD

Conceptual Overview

Classical Mechanics v.s. Neural Mechanics





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

Equation of learning:

Damped oscillator driven by loss gradients

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.

And the "nails"…

1. **Network Pruning**

   Hidenori Tanaka, Daniel Kunin*, Daniel LK Yamins, and Surya Ganguli.*
   Pruning neural networks without any data by iteratively conserving synaptic flow. NeurIPS 2020.

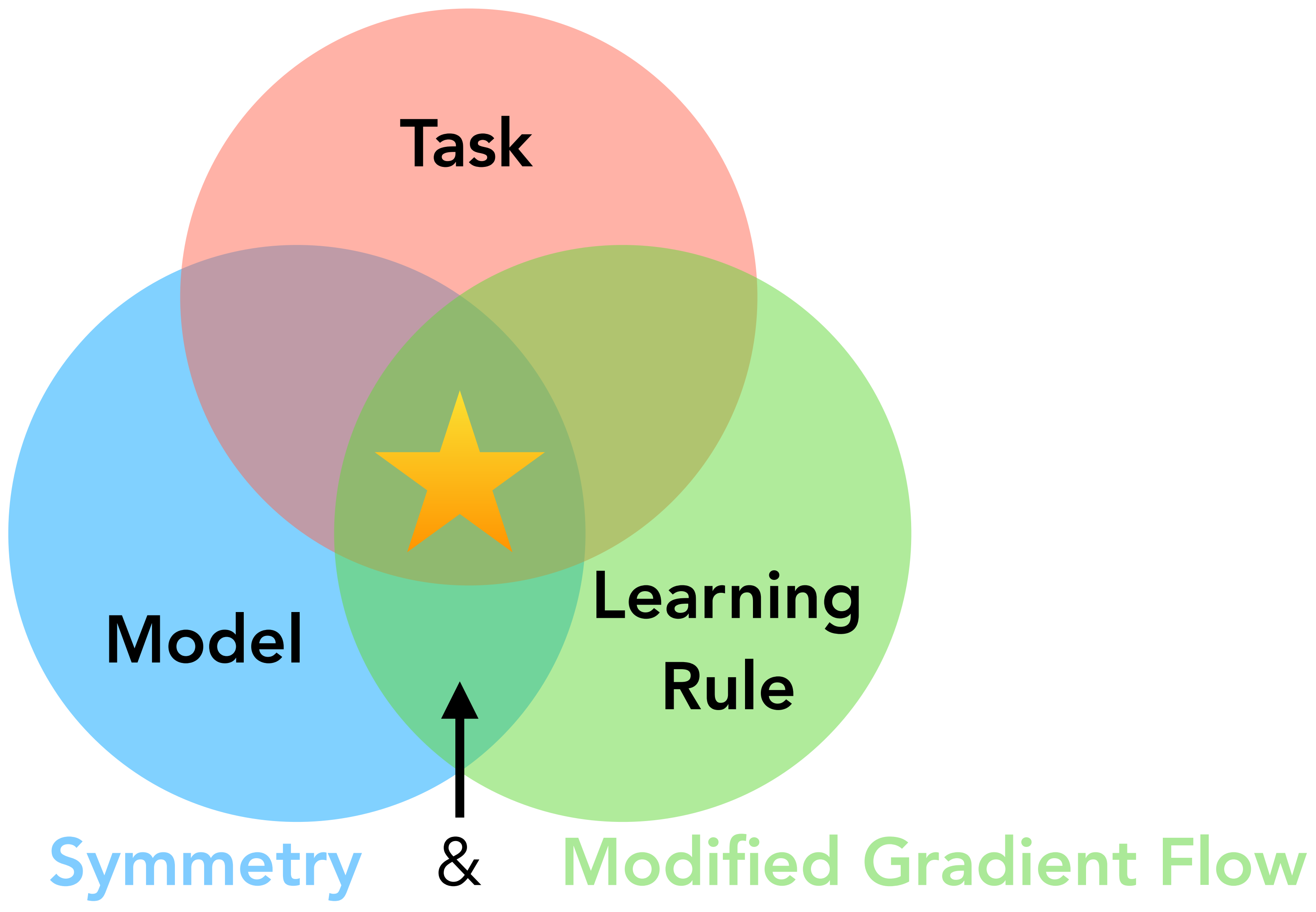
2. **Continual learning**

   Ekdeep Singh Lubana, Puja Trivedi, Robert P. Dick.
   Rethinking Quadratic Regularizers: Explicit Movement Regularization for Continual Learning. 2021

3. **Effective learning rate of BatchNorm**

   Zhiyuan Li, Sanjeev Arora
   An Exponential Learning Rate Schedule for Deep Learning. ICLR 2020

Where next?



Task

Model

Learning Rule

Symmetry & Modified Gradient Flow