Symmetries in AI4Science

Jan E. Gerken

CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF
GOTHENBURG

WASP
WALLENBERG AI,
AUTONOMOUS SYSTEMS
AND SOFTWARE PROGRAM

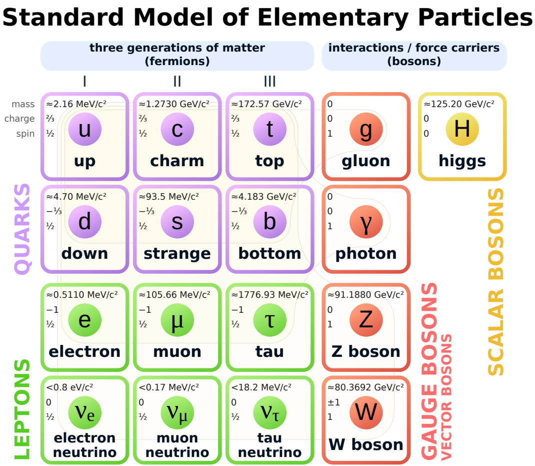
Workshop on Machine Learning Based Sampling in Lattice Field Theory and Quantum Chemistry

TRA Colloquium

Bonn
22th October 2024

Symmetries in physics

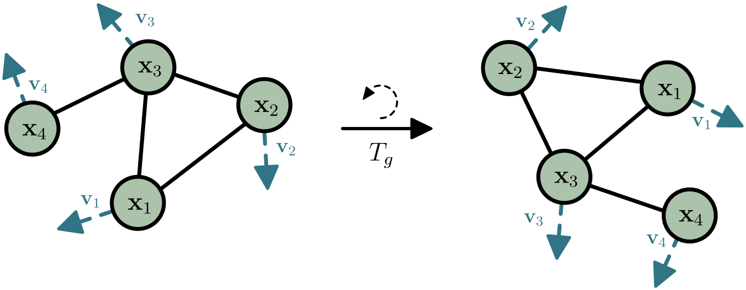
$$SU(2) \times SU(3) \times U(1) \qquad \longleftrightarrow$$



Standard Model of Elementary Particles

	three generations of matter (fermions)			interactions / force carriers (bosons)	
	I	II	III		
mass charge spin	≈2.16 MeV/c^2 ⅔ ½ u up	≈1.2730 GeV/c^2 ⅔ ½ c charm	≈172.57 GeV/c^2 ⅔ ½ t top	0 0 1 g gluon	≈125.20 GeV/c^2 0 0 H higgs
QUARKS	≈4.70 MeV/c^2 −⅓ ½ d down	≈93.5 MeV/c^2 −⅓ ½ s strange	≈4.183 GeV/c^2 −⅓ ½ b bottom	0 0 1 γ photon	SCALAR BOSONS
	≈0.5110 MeV/c^2 −1 ½ e electron	≈105.66 MeV/c^2 −1 ½ μ muon	≈1776.93 MeV/c^2 −1 ½ τ tau	≈91.1880 GeV/c^2 0 1 Z Z boson	GAUGE BOSONS VECTOR BOSONS
LEPTONS	<0.8 eV/c^2 0 ½ ν_e electron neutrino	<0.17 MeV/c^2 0 ½ ν_μ muon neutrino	<18.2 MeV/c^2 0 ½ ν_τ tau neutrino	≈80.3692 GeV/c^2 ±1 1 W W boson	

Symmetries in chemistry

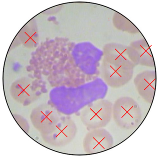


$$E \xrightarrow[T_g]{\circlearrowleft} E$$

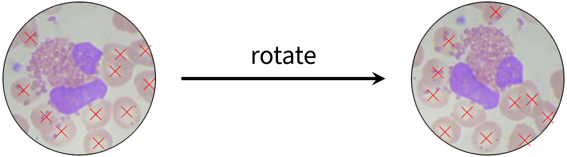
Image from: [Satorras et al. 2021]

Symmetries in prediction models

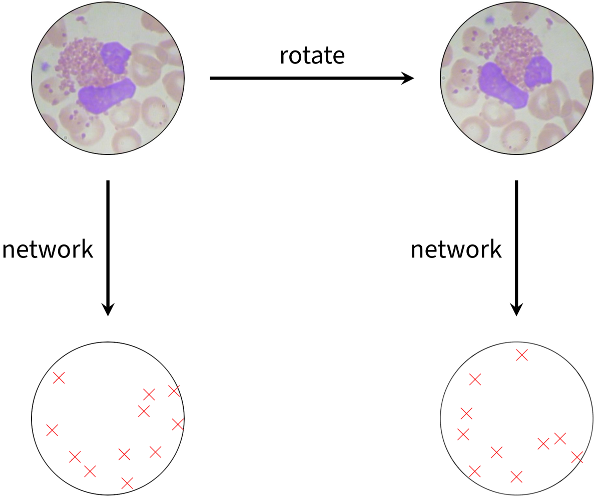
Symmetries in prediction models



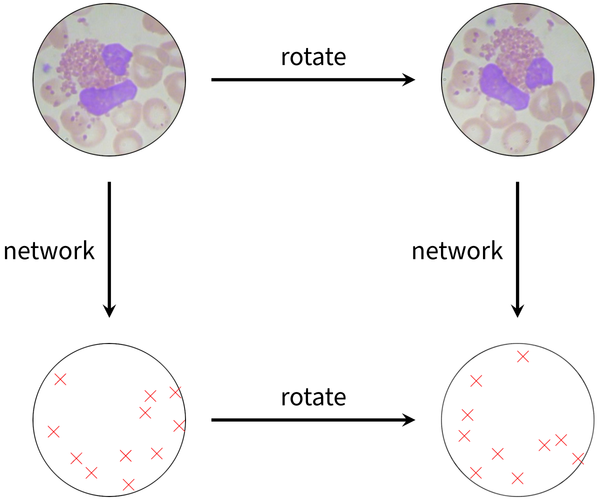
Symmetries in prediction models



Symmetries in prediction models



Symmetries in prediction models



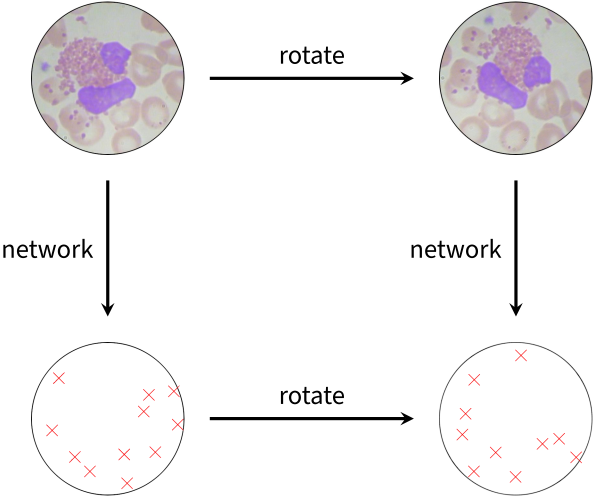
rotate

network

network

rotate

Symmetries in prediction models



rotate

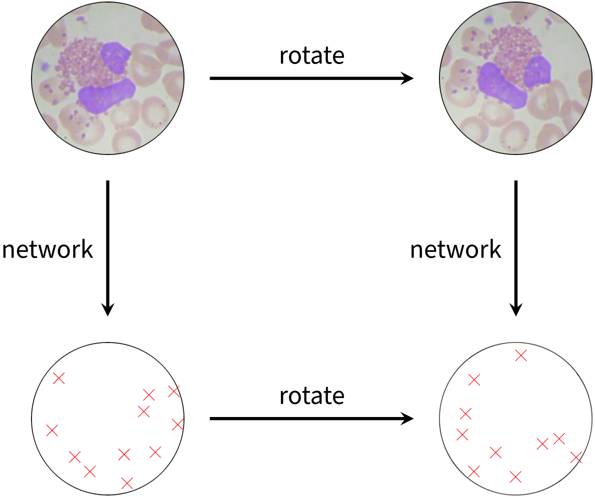
network

network

rotate

$$\iff \mathcal{N}(\rho_{\text{in}}(g)x) = \rho_{\text{out}}(g)\mathcal{N}(x)$$

Symmetries in prediction models



rotate

network

network

rotate

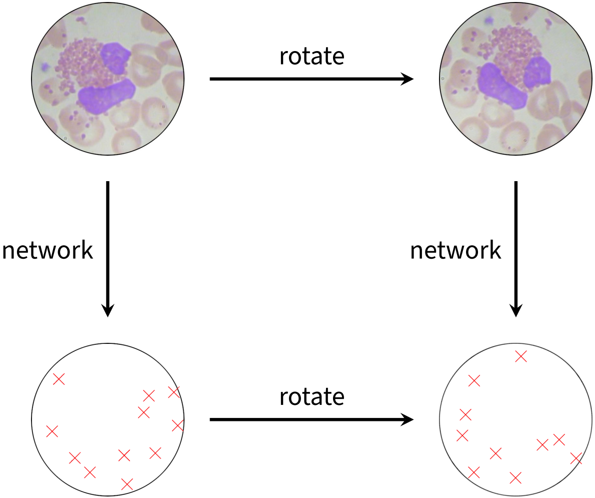
network

$$\Longleftrightarrow \quad \mathcal{N}(\rho_{\text{in}}(g)x) = \rho_{\text{out}}(g)\mathcal{N}(x)$$

rotation

image

Equivariance



rotate

network

network

rotate

$\iff$

network

$$\mathcal{N}(\rho_{\text{in}}(g)x) = \rho_{\text{out}}(g)\mathcal{N}(x)$$

rotation

image

Symmetries in generative models

- Sample from an invariant distribution

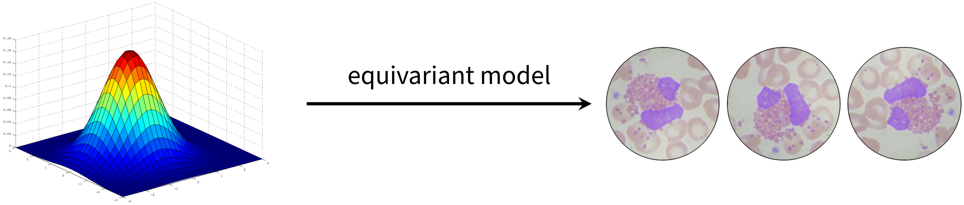
$$p(x) = p(\rho(g)x)$$

Symmetries in generative models

- Sample from an invariant distribution

$$p(x) = p(\rho(g)x)$$

- For latent variable models:



equivariant model

invariant latent distribution

Fundamental representation

- Groups act on vector spaces with representations

$$\rho : G \to \mathbb{R}^{n \times n}$$

Fundamental representation

- Groups act on vector spaces with representations

$$\rho : G \to \mathbb{R}^{n \times n}$$

- Vectors transform in the defining representation of matrix Lie groups

$$f_x \to \rho(g) f_x$$

Fundamental representation

- Groups act on vector spaces with representations

$$\rho : G \to \mathbb{R}^{n \times n}$$

- Vectors transform in the defining representation of matrix Lie groups

$$f_x \to \rho(g) f_x$$

- E.g. atom positions, force vectors

Regular representation

- A group acts on itself by left multiplication $h \to gh$

Regular representation

- A group acts on itself by left multiplication $h \to gh$

- This is leads to the regular representation on $\mathbb{R}^{|G|}$

$$\rho_{\text{reg}}(g)e_h = e_{gh}$$

Regular representation

- A group acts on itself by left multiplication $h \to gh$

- This is leads to the regular representation on $\mathbb{R}^{|G|}$

$$\rho_{\text{reg}}(g)e_h = e_{gh}$$

- Functions $G \to \mathbb{R}$ can be identified with $\mathbb{R}^{|G|}$

$$e_g \leftrightarrow \mathbb{1}_g$$

Regular representation

- A group acts on itself by left multiplication $h \to gh$

- This is leads to the regular representation on $\mathbb{R}^{|G|}$

$$\rho_{\text{reg}}(g)e_h = e_{gh}$$

- Functions $G \to \mathbb{R}$ can be identified with $\mathbb{R}^{|G|}$

$$e_g \leftrightarrow \mathbb{1}_g$$

- The regular representation on functions $f : G \to \mathbb{R}$ is given by

$$(\rho_{\text{reg}}(g)f)(h) = f(g^{-1}h)$$

Regular representation

- The regular representation on functions $f : G \to \mathbb{R}$ is given by

$$(\rho_{\text{reg}}(g)f)(h) = f(g^{-1}h)$$

Regular representation

- The regular representation on functions $f : G \to \mathbb{R}$ is given by

$$(\rho_{\text{reg}}(g)f)(h) = f(g^{-1}h)$$

- For functions on general domains, the regular representation is

$$(\rho_{\text{reg}}(g)f)(x) = f(\rho(g^{-1})x)$$

Regular representation

- The regular representation on functions $f : G \to \mathbb{R}$ is given by

$$(\rho_{\text{reg}}(g)f)(h) = f(g^{-1}h)$$

- For functions on general domains, the regular representation is

$$(\rho_{\text{reg}}(g)f)(x) = f(\rho(g^{-1})x)$$

- This is how scalar fields transform

Vector fields

- For functions $f : D \to \mathbb{R}^n$, combine defining- with regular representation

$$f(x) \to \pi(g) f(\rho^{-1}(g) x)$$

Vector fields

- For functions $f : D \to \mathbb{R}^n$, combine defining- with regular representation

$$f(x) \to \pi(g) f(\rho^{-1}(g) x)$$

- This is how vector fields transform

Vector fields

- For functions $f : D \to \mathbb{R}^n$, combine defining- with regular representation

$$f(x) \to \pi(g) f(\rho^{-1}(g) x)$$

- This is how vector fields transform

- π can also be other representation, e.g. adjoint representation

$$\pi(g) f = \rho(g)\, f\, \rho^{-1}(g)$$

Arbitrary representations

- Any finite-dimensional representation of a compact group can be written as a direct sum of irreducible representations

Arbitrary representations

- Any finite-dimensional representation of a compact group can be written as a direct sum of irreducible representations

- In particular, tensor product representations can be decomposed into direct sums

$$\rho^{\ell} \otimes \rho^{m} = \bigoplus_{n} (\rho^{n})^{\oplus c_{n}^{\ell m}}$$

Arbitrary representations

- Any finite-dimensional representation of a compact group can be written as a direct sum of irreducible representations

- In particular, tensor product representations can be decomposed into direct sums

$$\rho^{\ell} \otimes \rho^{m} = \bigoplus_{n} (\rho^{n})^{\oplus c_n^{\ell m}}$$

- Change of basis done via Clebsch–Gordan coefficients

How to construct equivariant layers?

- Consider linear maps M which are equivariant

How to construct equivariant layers?

- Consider linear maps *M* which are equivariant

- For vectors, *M* needs to be an intertwiner

$$M\rho_{\text{in}}(g) = \rho_{\text{out}}(g)M \qquad \forall g \in G \qquad (*)$$

How to construct equivariant layers?

- Consider linear maps M which are equivariant

- For vectors, M needs to be an intertwiner

$$M\rho_{\text{in}}(g) = \rho_{\text{out}}(g)M \qquad \forall g \in G \qquad\qquad (*)$$

- Schur's lemma: for complex, irreducible representations

$$M = \lambda\mathbb{1} \quad \text{if} \quad \rho_{\text{in}} = \rho_{\text{out}} \qquad \text{and} \qquad M = 0 \quad \text{otherwise}$$

How to construct equivariant layers?

- Consider linear maps M which are equivariant
- For vectors, M needs to be an intertwiner

$$M\rho_{\text{in}}(g) = \rho_{\text{out}}(g)M \qquad \forall g \in G \tag{$*$}$$

- Schur's lemma: for complex, irreducible representations

$$M = \lambda\mathbb{1} \quad \text{if} \quad \rho_{\text{in}} = \rho_{\text{out}} \qquad \text{and} \qquad M = 0 \quad \text{otherwise}$$

- Hence, decompose ρ_{in}, ρ_{out} into irreps to solve $(*)$

How to construct equivariant layers?

- For the regular representation, linear equivariant layers are given by group convolutions [Cohen, Welling 2016]

$$[\psi * f](g) = \int_G \mathrm{d}h \, \psi(h^{-1}g) f(h)$$

How to construct equivariant layers?

- For the regular representation, linear equivariant layers are given by group convolutions [Cohen, Welling 2016]

$$[\psi * f](g) = \int_G \mathrm{d}h\, \psi(h^{-1}g) f(h)$$

- For the translation group, these become the usual convolutions

How to construct equivariant layers?

- For the regular representation, linear equivariant layers are given by group convolutions [Cohen, Welling 2016]

$$[\psi \ast f](g) = \int_G \mathrm{d}h\, \psi(h^{-1}g) f(h)$$

- For the translation group, these become the usual convolutions

- For combination with fundamental representation $(\pi(g) f(\rho^{-1}(g)x))$, convolution filter needs to be an intertwiner

[Review: Weiler et al. 2023]

Equivariance in quantum chemistry

- Group:
  roto-translations of the molecule + permutations of identical atoms

Equivariance in quantum chemistry

- Group:
  roto-translations of the molecule + permutations of identical atoms

- Use graph-NNs for permutation part

Equivariance in quantum chemistry

- Group:
  roto-translations of the molecule + permutations of identical atoms

- Use graph-NNs for permutation part

- For SO(3), expand in irreps, use tensor products to combine features

[Review: Duval et al. 2023]

Gauge symmetry

- In a gauge symmetry, group element can depend on position x

$$\pi(g(x))f(\rho^{-1}(g(x))x)$$

Gauge symmetry

- In a gauge symmetry, group element can depend on position *x*

$$\pi(g(x))f(\rho^{-1}(g(x))x)$$

- Symmetry becomes local

Gauge symmetry

- In a gauge symmetry, group element can depend on position x

$$\pi(g(x))f(\rho^{-1}(g(x))x)$$

- Symmetry becomes local

- In quantum chemistry only global symmetries

Gauge symmetry

- In a gauge symmetry, group element can depend on position x

$$\pi(g(x))f(\rho^{-1}(g(x))x)$$

- Symmetry becomes local

- In quantum chemistry only global symmetries

- Equivariance wrt local coordinate changes is also a gauge symmetry: Gauge CNNs [Cheng et al. 2019]

Equivariance in lattice field theory

- In lattice field theory, typically combination of local and global symmetries: $G = SU(n) \times SE(3)$

$$\pi(g(x))f(\rho^{-1}(h)x) \qquad g(x) \in \mathrm{SU}(n) \qquad h \in \mathrm{SE}(3)$$

Equivariance in lattice field theory

- In lattice field theory, typically combination of local and global symmetries: $G = SU(n) \times SE(3)$

$$\pi(g(x))f(\rho^{-1}(h)x) \qquad g(x) \in \mathrm{SU}(n) \qquad h \in \mathrm{SE}(3)$$

- The gauge group acts in the adjoint representation

$$\pi(g(x))f = \rho(g(x))\, f\, \rho^{\dagger}(g(x))$$

Equivariance in lattice field theory

- In lattice field theory, typically combination of local and global symmetries: $G = SU(n) \times SE(3)$

$$\pi(g(x))f(\rho^{-1}(h)x) \qquad g(x) \in \mathrm{SU}(n) \qquad h \in \mathrm{SE}(3)$$

- The gauge group acts in the adjoint representation

$$\pi(g(x))f = \rho(g(x))\, f\, \rho^{\dagger}(g(x))$$

- By discretizing on the lattice, obtain links U_μ transforming as

$$U_\mu(x) \to \rho(g(x))U_\mu(x)\, \rho^{\dagger}(g(x+\hat{\mu}))$$

How to construct gauge equivariant layers?

- Can build loops transforming as

$$W(x) \to \rho(g(x)) W(x)\, \rho^\dagger(g(x)) \qquad (*)$$

How to construct gauge equivariant layers?

- Can build loops transforming as

$$W(x) \to \rho(g(x))W(x)\,\rho^\dagger(g(x)) \qquad (*)$$

- Products of loops are equivariant, traces are invariant

$$\begin{aligned} W(x)\tilde{W}(x) &\to \rho(g(x))W(x)\tilde{W}(x)\,\rho^\dagger(g(x)) \\ \mathrm{tr}(W(x)) &\to \mathrm{tr}(W(x)) \end{aligned}$$

How to construct gauge equivariant layers?

- Can build loops transforming as

$$W(x) \to \rho(g(x))W(x)\,\rho^\dagger(g(x)) \qquad (*)$$

- Products of loops are equivariant, traces are invariant

$$\begin{aligned} W(x)\tilde{W}(x) &\to \rho(g(x))W(x)\tilde{W}(x)\,\rho^\dagger(g(x)) \\ \mathrm{tr}(W(x)) &\to \mathrm{tr}(W(x)) \end{aligned}$$

- Use this together with convolutions to build gauge equivariant networks

[Favoni et al. 2020]

How to construct gauge equivariant layers?

- Can build loops transforming as

$$W(x) \to \rho(g(x))W(x)\,\rho^\dagger(g(x)) \qquad (\ast)$$

- Products of loops are equivariant, traces are invariant

$$\begin{aligned} W(x)\tilde{W}(x) &\to \rho(g(x))W(x)\tilde{W}(x)\,\rho^\dagger(g(x)) \\ \mathrm{tr}(W(x)) &\to \mathrm{tr}(W(x)) \end{aligned}$$

- Use this together with convolutions to build gauge equivariant networks [Favoni et al. 2020]

- Can also manipulate invariants of $(\ast)$ [Boyda et al. 2021]

How to construct gauge equivariant layers?

- Can build loops transforming as

$$W(x) \to \rho(g(x))W(x)\,\rho^\dagger(g(x)) \qquad (*)$$

- Products of loops are equivariant, traces are invariant

$$W(x)\tilde{W}(x) \to \rho(g(x))W(x)\tilde{W}(x)\,\rho^\dagger(g(x))$$
$$\mathrm{tr}(W(x)) \to \mathrm{tr}(W(x))$$

- Use this together with convolutions to build gauge equivariant networks [Favoni et al. 2020]
- Can also manipulate invariants of $(*)$ [Boyda et al. 2021]
- Can differentiate an invariant [Bacchio et al. 2023]

Part II: Other ways of reaching equivariance

(Frame) averaging

- Create exactly equivariant model by averaging over the group

$$\bar{f}(x) = \int_G \mathrm{d}h \, \pi(g) f(\rho^{-1}(g)x)$$

(Frame) averaging

- Create exactly equivariant model by averaging over the group

$$\bar{f}(x) = \int_G \mathrm{d}h \, \pi(g) f(\rho^{-1}(g) x)$$

👍 It is sufficient to average over an equivariant subset $\mathcal{F}(x) \subset G$ (frame)

[Puny et al. 2022]

(Frame) averaging

- Create exactly equivariant model by averaging over the group

$$\bar{f}(x) = \int_G \mathrm{d}h\, \pi(g) f(\rho^{-1}(g)x)$$

👍 It is sufficient to average over an equivariant subset $\mathcal{F}(x) \subset G$ (frame)

[Puny et al. 2022]

👍 Works with any architecture

(Frame) averaging

- Create exactly equivariant model by averaging over the group

$$\bar{f}(x) = \int_G \mathrm{d}h \, \pi(g) f(\rho^{-1}(g)x)$$

👍 It is sufficient to average over an equivariant subset $\mathcal{F}(x) \subset G$ (frame) [Puny et al. 2022]

👍 Works with any architecture

👎 Only approximate for continuous groups when sampling is necessary to evaluate the integral

Canonicalization

[Kaba et al. 2022]

- Use an equivariant map $D \to G$ to predict a canonicalizing transformation

Canonicalization

[Kaba et al. 2022]

- Use an equivariant map $D \to G$ to predict a canonicalizing transformation

- Use non-equivariant network for prediction

Canonicalization

[Kaba et al. 2022]

- Use an equivariant map $D \rightarrow G$ to predict a canonicalizing transformation

- Use non-equivariant network for prediction

👍 Exactly equivariant

Canonicalization

[Kaba et al. 2022]

- Use an equivariant map $D \to G$ to predict a canonicalizing transformation

- Use non-equivariant network for prediction

👍 Exactly equivariant

👎 Still needs equivariant model

Canonicalization

[Kaba et al. 2022]

- Use an equivariant map $D \to G$ to predict a canonicalizing transformation

- Use non-equivariant network for prediction

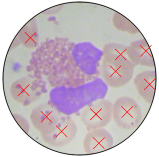
👍 Exactly equivariant

👎 Still needs equivariant model

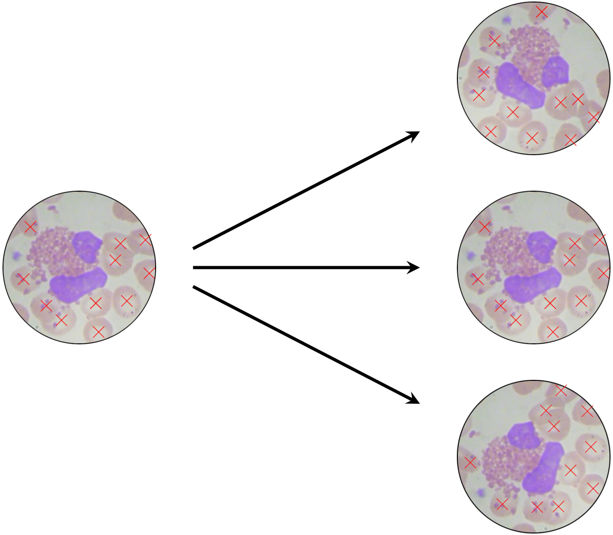
👎 Equivariant function with codomain G is hard to construct

Data augmentation

Data augmentation



Data augmentation



Data augmentation

- 👍 Easy to implement

- 👍 No specialized architecture necessary

Data augmentation

👍 Easy to implement

👍 No specialized architecture necessary

👎 No exact equivariance

Data augmentation

👍 Easy to implement

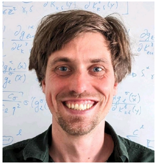
👍 No specialized architecture necessary

👎 No exact equivariance

Can we understand data augmentation theoretically?

Emergent Equivariance in Deep Ensembles

in collaboration with



Pan Kessel

Empirical NTK

Training dynamics under continuous gradient descent:

learning rate

loss

$$\frac{\mathrm{d}\mathcal{N}_\theta(x)}{\mathrm{d}t} = -\frac{\eta}{N}\sum_{i=1}^{N} \Theta_\theta(x, x_i)\,\frac{\partial L}{\partial \mathcal{N}(x_i)}$$

training sample

Empirical NTK

Training dynamics under continuous gradient descent:

learning rate

loss

$$\frac{\mathrm{d}\mathcal{N}_{\theta}(x)}{\mathrm{d}t} = -\frac{\eta}{N}\sum_{i=1}^{N} \Theta_{\theta}(x, x_i)\frac{\partial L}{\partial \mathcal{N}(x_i)}$$

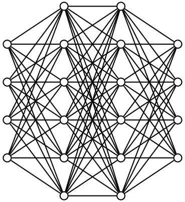
training sample

with the empirical neural tangent kernel (NTK)

$$\Theta_{\theta}(x, x') = \sum_{\mu} \frac{\partial \mathcal{N}(x)}{\partial \theta_{\mu}}\frac{\partial \mathcal{N}(x')}{\partial \theta_{\mu}}$$

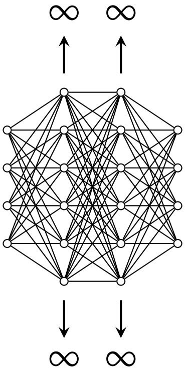
Infinite width limit

[Jacot et al. 2018]



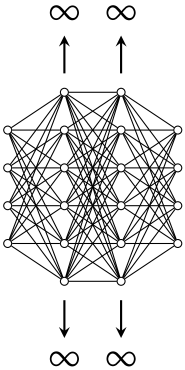
Infinite width limit

[Jacot et al. 2018]



Infinite width limit

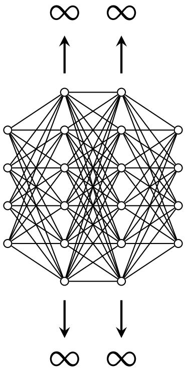
[Jacot et al. 2018]



👍 NTK becomes independent of initialization

Infinite width limit

[Jacot et al. 2018]

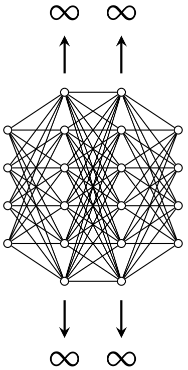


👍 NTK becomes independent of initialization

👍 NTK becomes constant in training

Infinite width limit

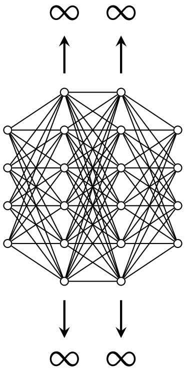
[Jacot et al. 2018]



- 👍 NTK becomes independent of initialization
- 👍 NTK becomes constant in training
- 👍 NTK can be computed for most networks

Infinite width limit

[Jacot et al. 2018]



👍 NTK becomes independent of initialization

👍 NTK becomes constant in training

👍 NTK can be computed for most networks

✓ Training dynamics can be solved

Mean prediction from NTK

[Jacot et al. 2018]

- At infinite width, the mean prediction is given by

$$\mu_t(x) = \Theta(x, X)\Theta(X, X)^{-1}(\mathbb{I} - e^{-\eta\Theta(X,X)t})Y$$

Mean prediction from NTK

[Jacot et al. 2018]

At infinite width, the mean prediction is given by

$$\mu_t(x) = \overset{\text{neural tangent kernel}}{\Theta}(x, X)\Theta(X, X)^{-1}(\mathbb{I} - e^{-\eta\Theta(X,X)t})Y$$

Mean prediction from NTK

[Jacot et al. 2018]

At infinite width, the mean prediction is given by

$$\mu_t(x) = \overset{\text{neural tangent kernel}}{\Theta}(x, \underset{\text{train data}}{X})\Theta(X,X)^{-1}(\mathbb{I} - e^{-\eta\Theta(X,X)t})Y$$

Mean prediction from NTK

[Jacot et al. 2018]

At infinite width, the mean prediction is given by

neural tangent kernel

$$\mu_t(x) = \Theta(x, X)\Theta(X, X)^{-1}(\mathbb{I} - e^{-\eta\Theta(X,X)t})Y$$

learning rate

train data

Mean prediction from NTK

[Jacot et al. 2018]

At infinite width, the mean prediction is given by

$$\mu_t(x) = \overset{\text{neural tangent kernel}}{\Theta}(x, \underset{\text{train data}}{X})\Theta(X,X)^{-1}(\mathbb{I} - e^{-\underset{\text{learning rate}}{\eta}\Theta(X,X)t})\overset{\text{train labels}}{Y}$$

Data augmentation

Data augmentation at infinite width

$$\mu_t(x) = \Theta(x, X)\Theta(X, X)^{-1}(\mathbb{I} - e^{-\eta\Theta(X,X)t})Y$$

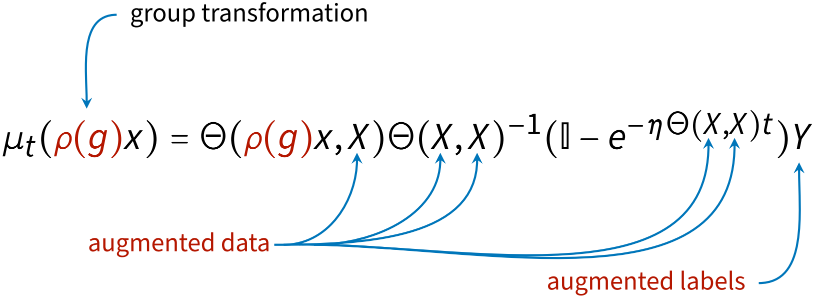
Data augmentation at infinite width

$$\mu_t(x) = \Theta(x, X)\Theta(X, X)^{-1}(\mathbb{I} - e^{-\eta\Theta(X,X)t})Y$$

augmented data

augmented labels

Data augmentation at infinite width



group transformation

$$\mu_t(\rho(g)x) = \Theta(\rho(g)x, X)\Theta(X, X)^{-1}(\square - e^{-\eta\Theta(X,X)t})Y$$

augmented data

augmented labels

Data augmentation at infinite width

group transformation

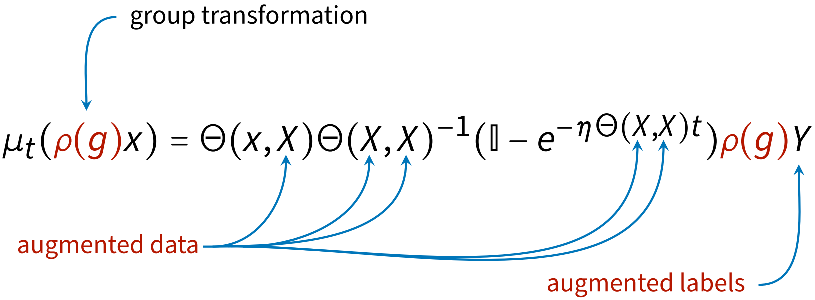
for augmented data

$$\mu_t(\rho(g)x) = \Theta(\rho(g)x, X)\Theta(X, X)^{-1}(\mathbb{1} - e^{-\eta\Theta(X,X)t})Y$$

augmented data

augmented labels

Data augmentation at infinite width



group transformation

$$\mu_t(\rho(g)x) = \Theta(x, X)\Theta(X, X)^{-1}(\mathbb{I} - e^{-\eta\Theta(X,X)t})\rho(g)Y$$

augmented data

augmented labels

Data augmentation at infinite width

group transformation

augmented labels

$$\mu_t(\rho(g)x) = \Theta(x,X)\Theta(X,X)^{-1}(\mathbb{I} - e^{-\eta\Theta(X,X)t})\underbrace{\rho(g)Y}_{=Y}$$

for invariance

Data augmentation at infinite width

group transformation

$$\begin{aligned}\mu_t(\rho(g)x) &= \Theta(x,X)\Theta(X,X)^{-1}(\mathbb{1} - e^{-\eta\Theta(X,X)t})\underbrace{\rho(g)Y}_{\substack{=Y\\ \text{for invariance}}}\\ &= \mu_t(x)\end{aligned}$$

Mean prediction

$$\mu_t(x)$$

Mean prediction

$$\mu_t(x) = \mathbb{E}_{\theta_0 \sim \text{initializations}}[\mathcal{N}_{\theta_t}(x)]$$

Mean prediction

$$\mu_t(x) = \mathbb{E}_{\theta_0 \sim \text{initializations}}[\mathcal{N}_{\theta_t}(x)] = \lim_{n \to \infty} \frac{1}{n} \sum_{\theta_0 = \text{init}_1}^{\text{init}_n} \mathcal{N}_{\theta_t}(x)$$

Mean prediction

$$\mu_t(x) = \mathbb{E}_{\theta_0 \sim \text{initializations}}[\mathcal{N}_{\theta_t}(x)] = \lim_{n \to \infty} \underbrace{\frac{1}{n} \sum_{\theta_0 = \text{init}_1}^{\text{init}_n} \mathcal{N}_{\theta_t}(x)}_{\text{mean prediction of deep ensemble}}$$

Main conclusion

Deep ensembles trained with data augmentation are equivariant.

Main conclusion

Deep ensembles trained with data augmentation are equivariant.

✓ Proof of exact equivariance for

- full data augmentation
- infinite ensembles
- at infinite width

Main conclusion

Deep ensembles trained with data augmentation are equivariant.

✓ Proof of exact equivariance for

- full data augmentation
- infinite ensembles
- at infinite width

✓ Equivariance holds for all training times

Main conclusion

Deep ensembles trained with data augmentation are equivariant.

✓ Proof of exact equivariance for

- full data augmentation
- infinite ensembles
- at infinite width

✓ Equivariance holds for all training times

✓ Equivariance holds away from the training data

Intuitive explanation

- ✓ Equivariance holds for all training times
- ✓ Equivariance holds away from the training data

Intuitive explanation

- ✓ Equivariance holds for all training times
- ✓ Equivariance holds away from the training data

ⓘ At infinite width, the mean output at initialization is zero everywhere.

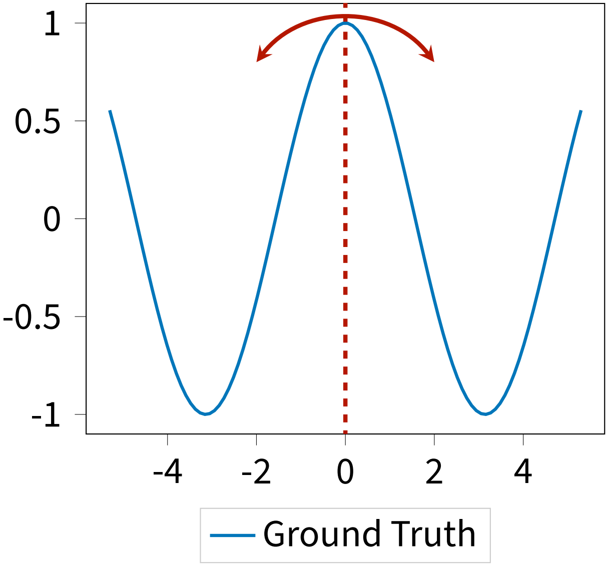
Intuitive explanation

- ✓ Equivariance holds for all training times
- ✓ Equivariance holds away from the training data

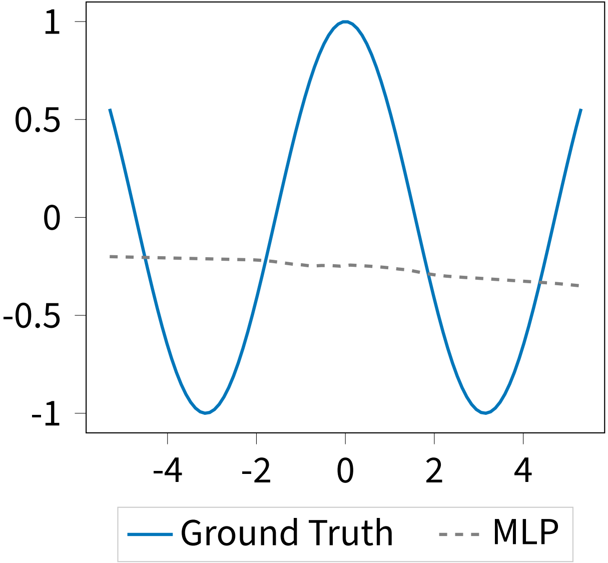
ⓘ At infinite width, the mean output at initialization is zero everywhere.

⇨ Training with full data augmentation leads to an equivariant function.

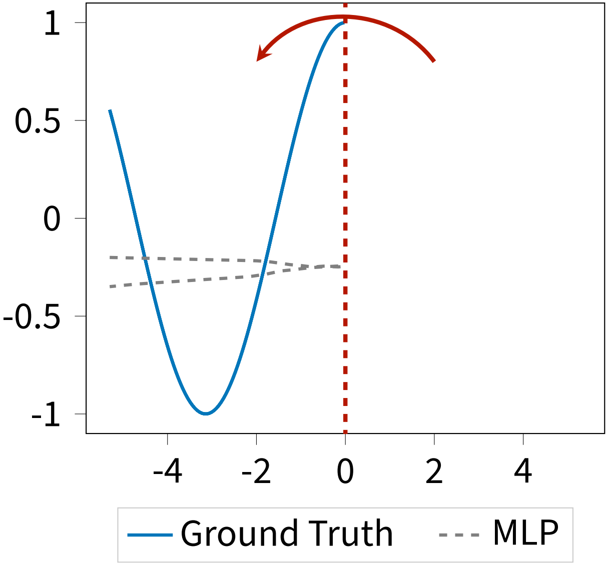
Toy example


1
0.5
0
-0.5
-1
-4
-2
0
2
4
Ground Truth

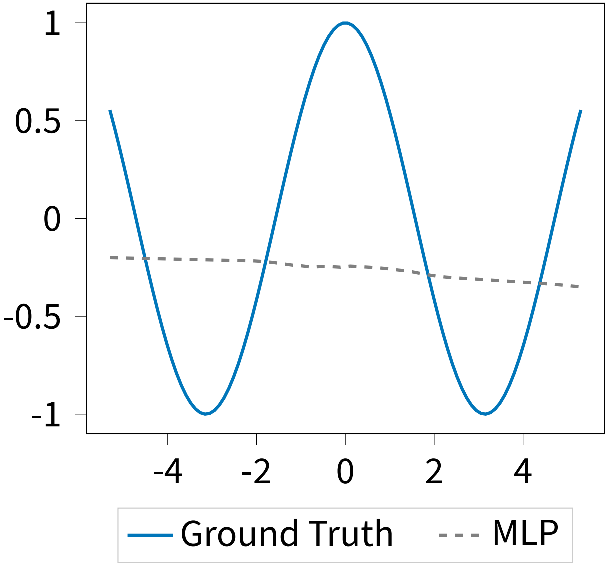
Initialization



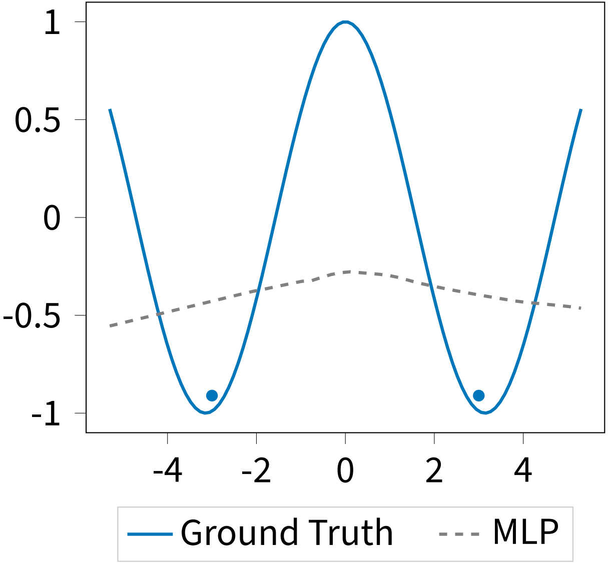
Initialization



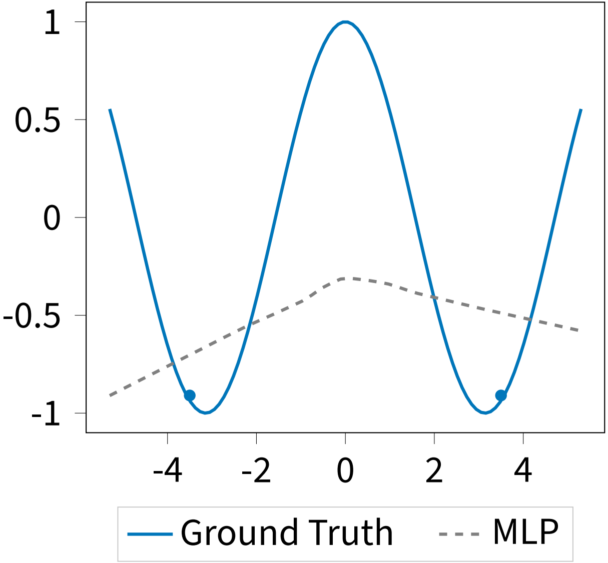
Initialization



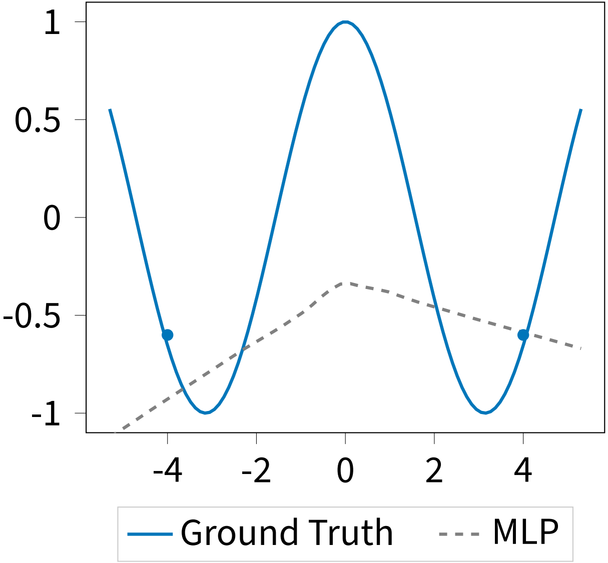
After 1 Training Step



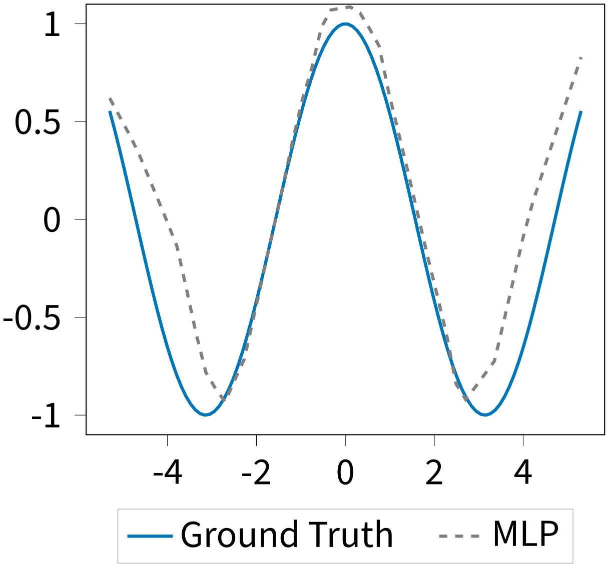
After 2 Training Steps



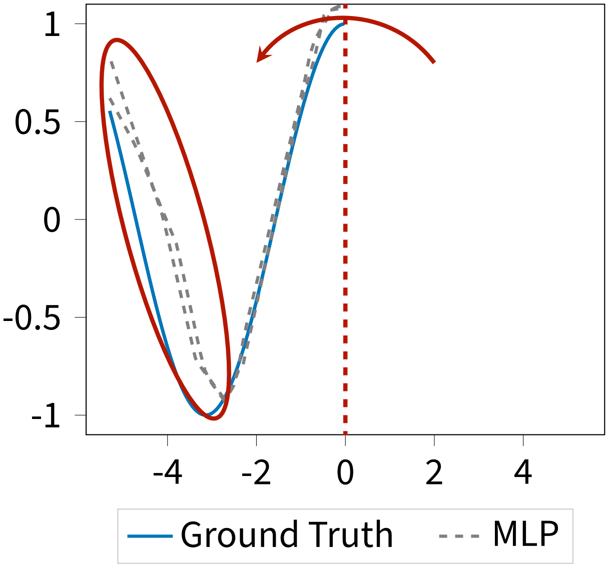
After 3 Training Steps



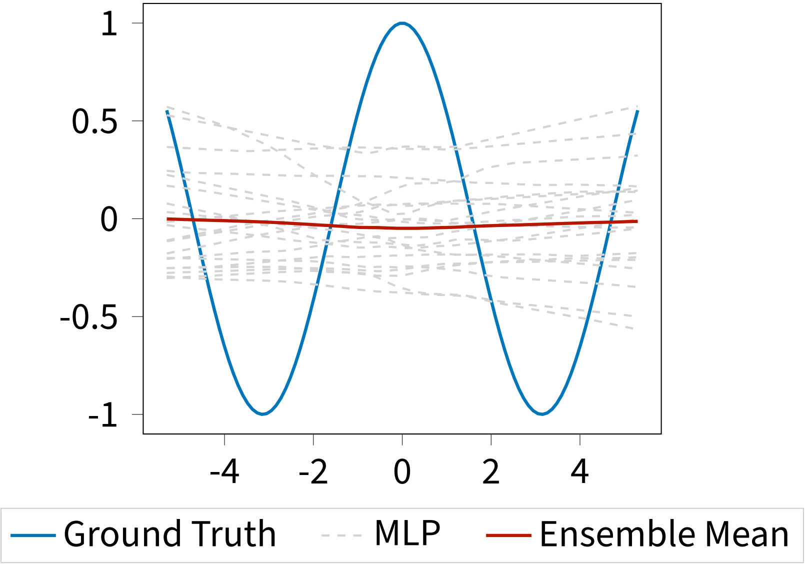
After 2000 Training Steps



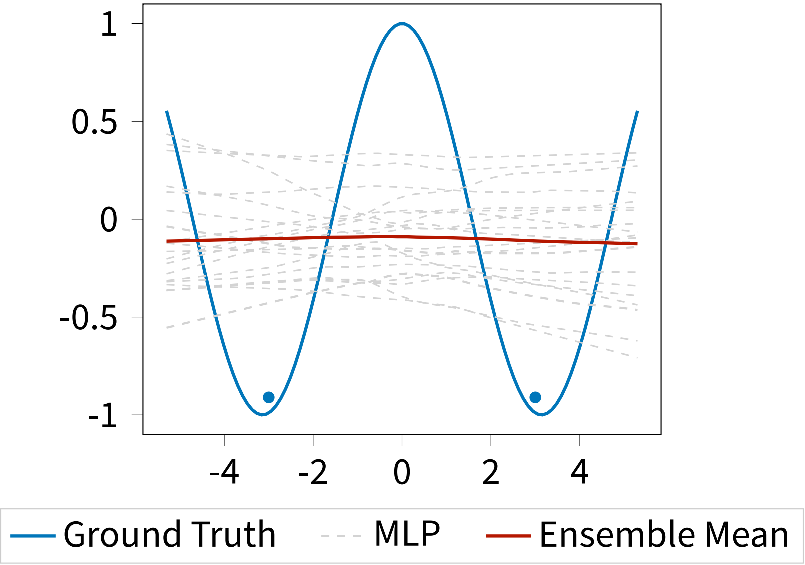
After 2000 Training Steps



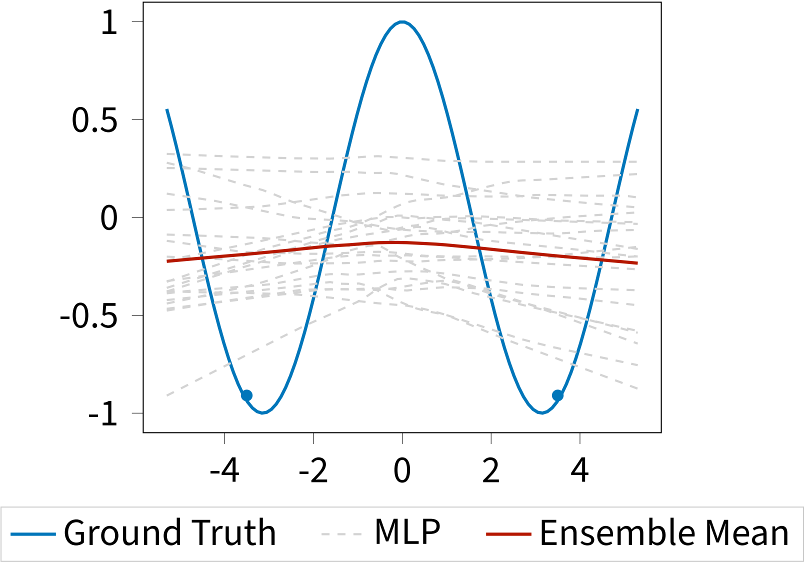
Initialization



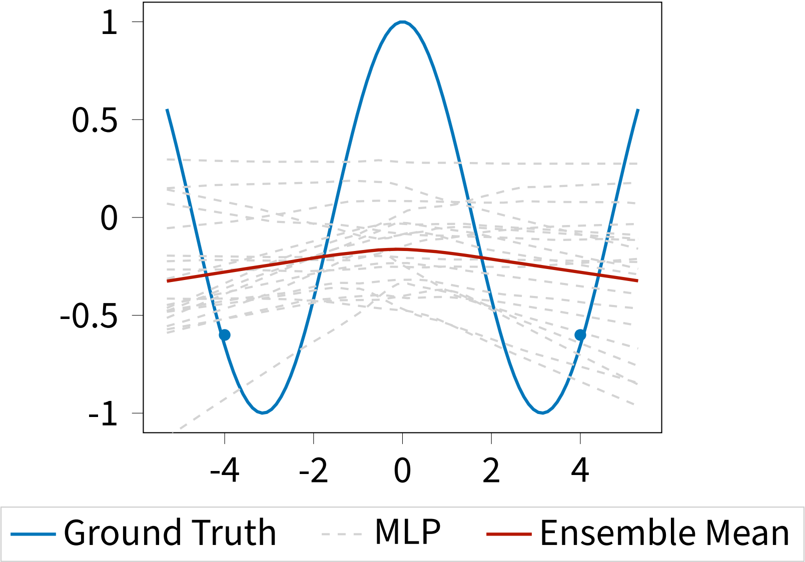
After 1 Training Step



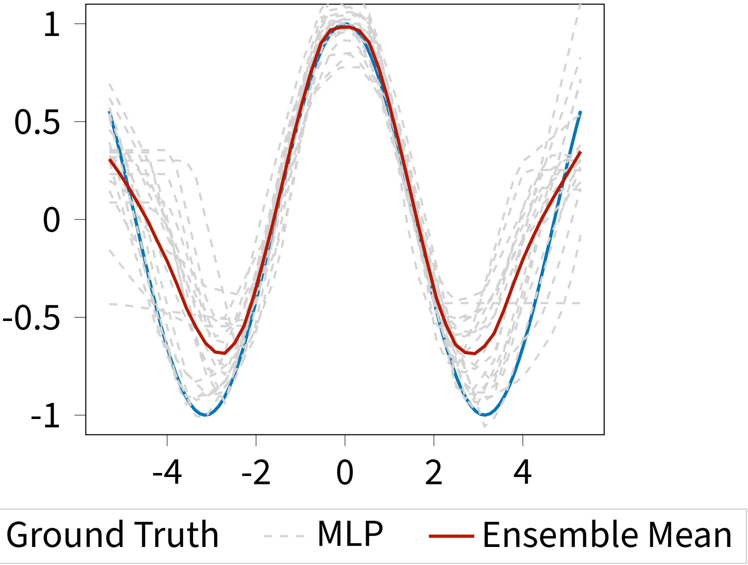
After 2 Training Steps



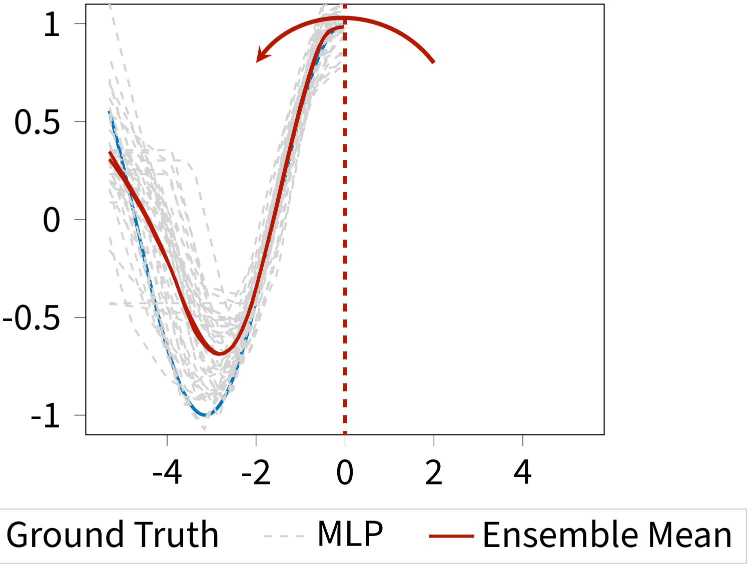
After 3 Training Steps



After 2000 Training Steps

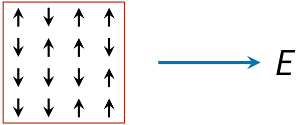


After 2000 Training Steps

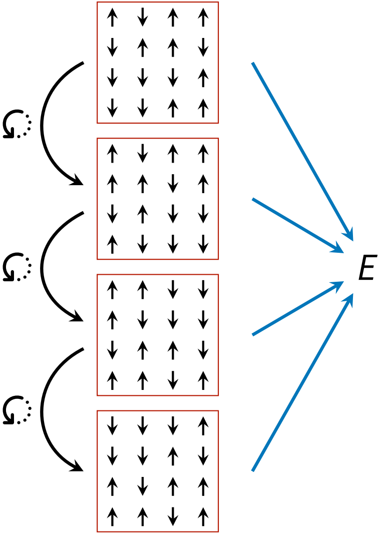


Experiments

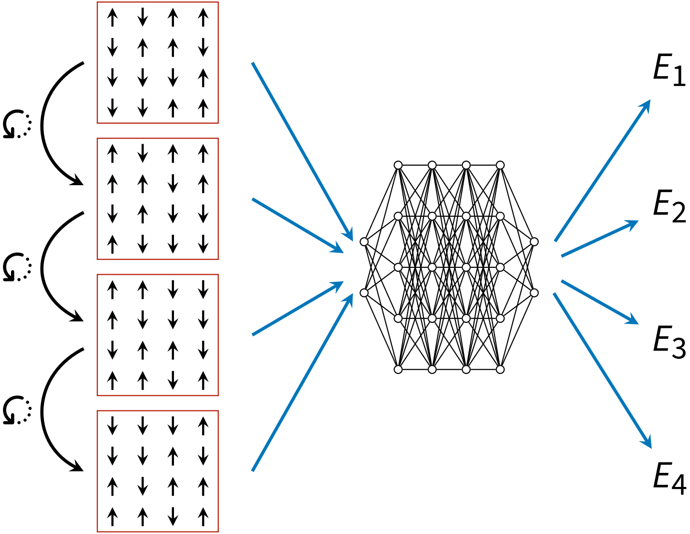
Ising model



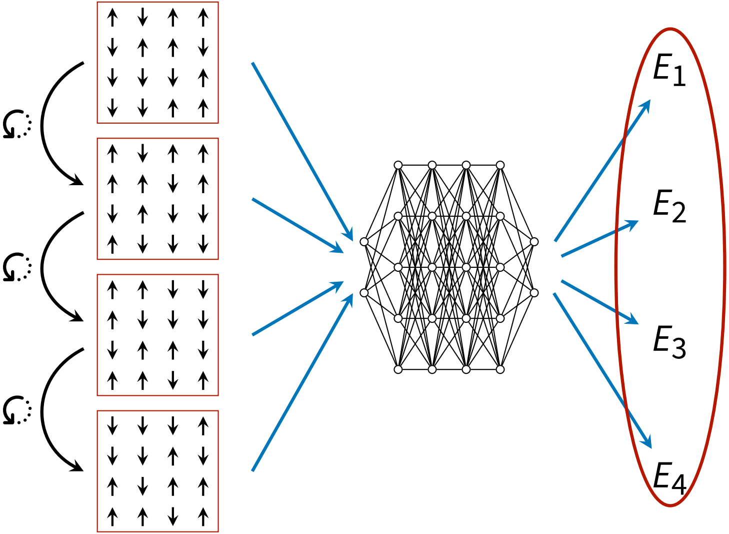
Ising model



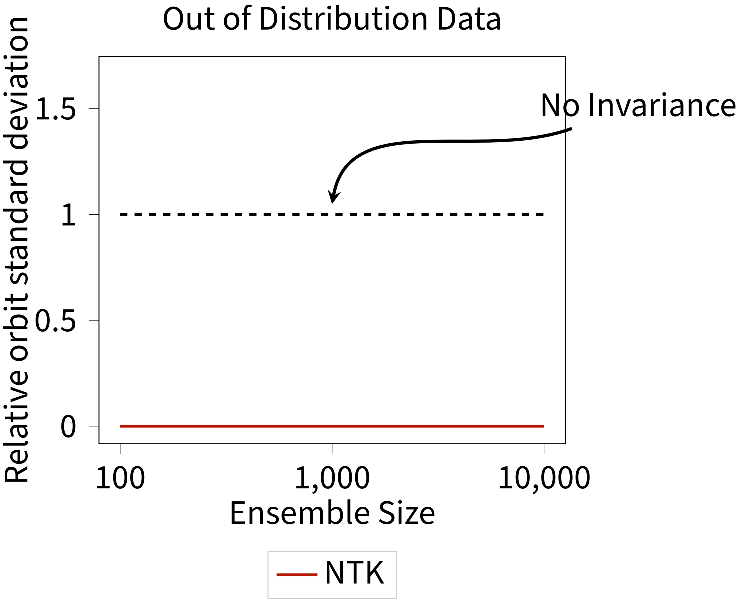
Ising model

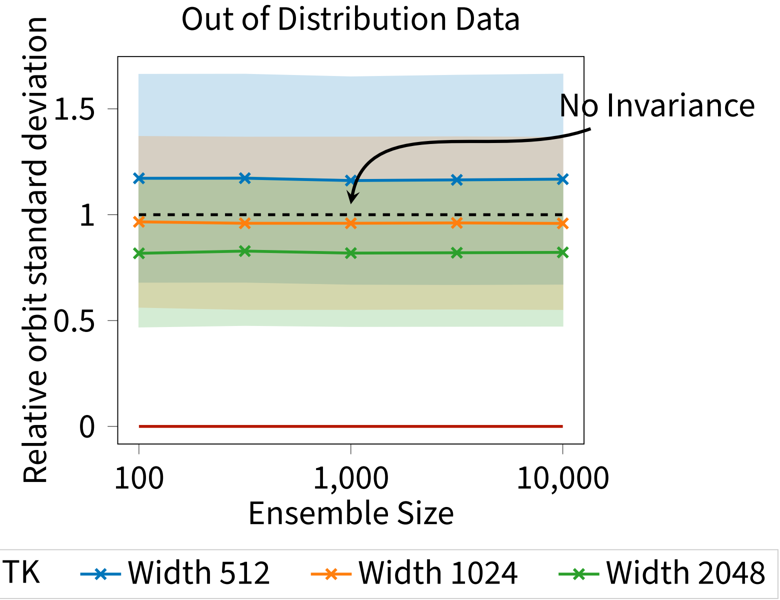


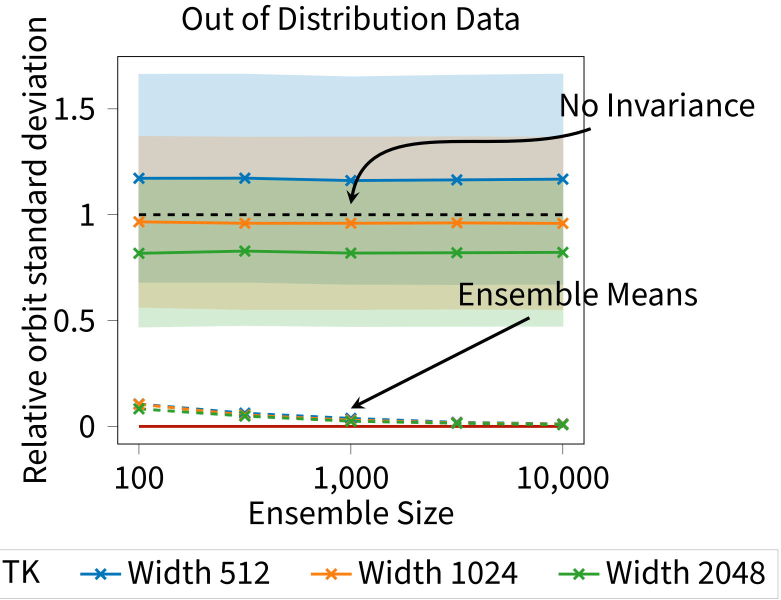
Ising model

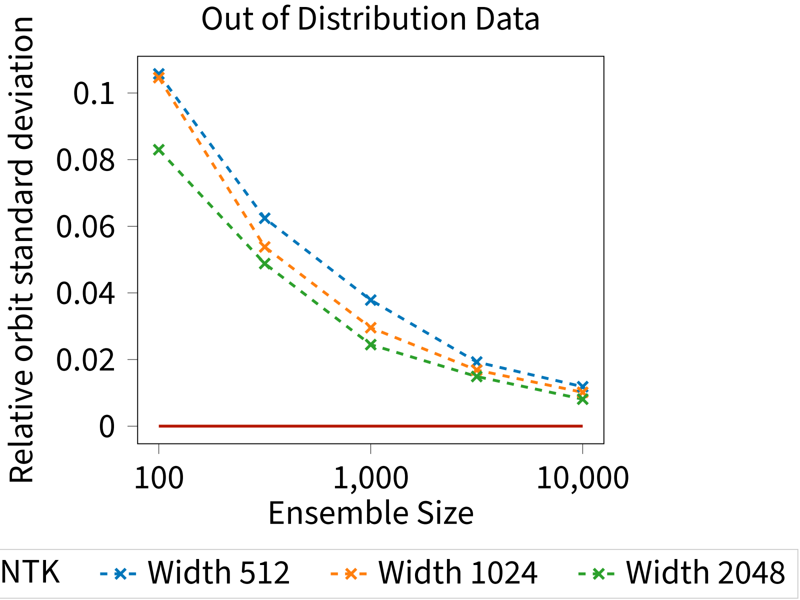


Relative Standard Deviation


Out of Distribution Data
No Invariance
Relative orbit standard deviation
1.5
1
0.5
0
100
1,000
10,000
Ensemble Size
NTK


Out of Distribution Data
No Invariance
Relative orbit standard deviation
1.5
1
0.5
0
100
1,000
10,000
Ensemble Size
NTK
Width 512
Width 1024
Width 2048

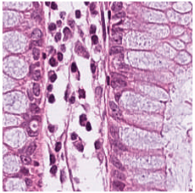

Out of Distribution Data
Relative orbit standard deviation
1.5
1
0.5
0
100
1,000
10,000
Ensemble Size
No Invariance
Ensemble Means
NTK
Width 512
Width 1024
Width 2048



Out of Distribution Data
Relative orbit standard deviation
0.1
0.08
0.06
0.04
0.02
0
100
1,000
10,000
Ensemble Size
NTK
Width 512
Width 1024
Width 2048

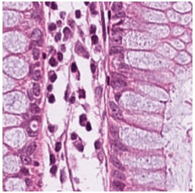
Histological slices

[Kather et al. 2018]



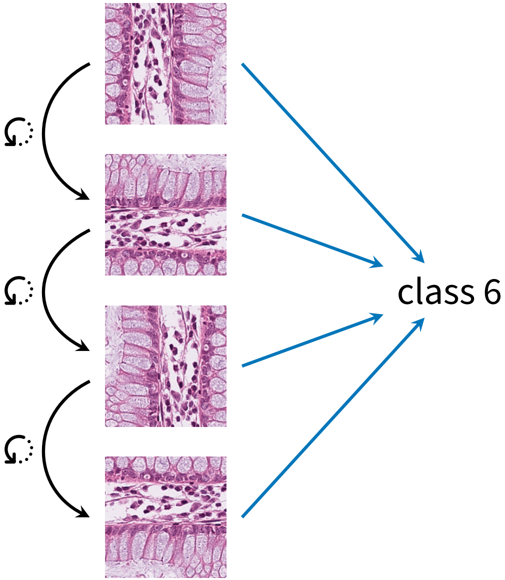
Histological slices

[Kather et al. 2018]

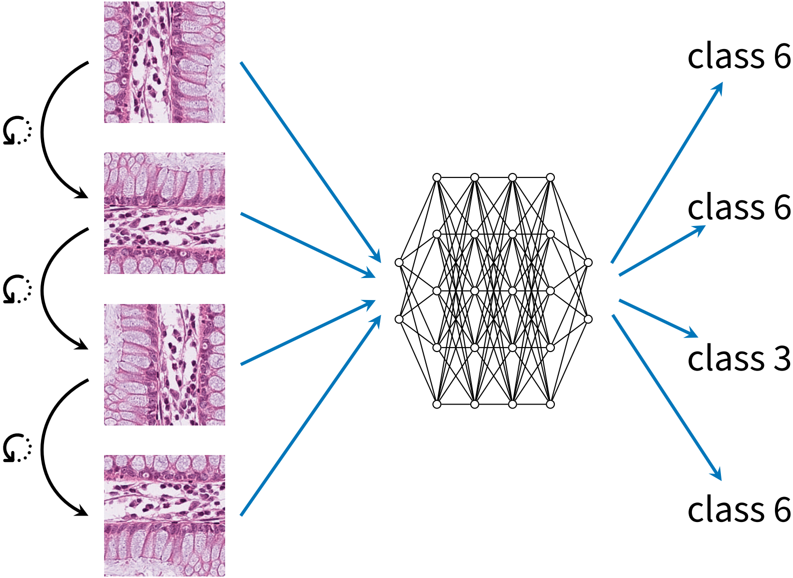


→ class 6

Histological slices



Histological slices



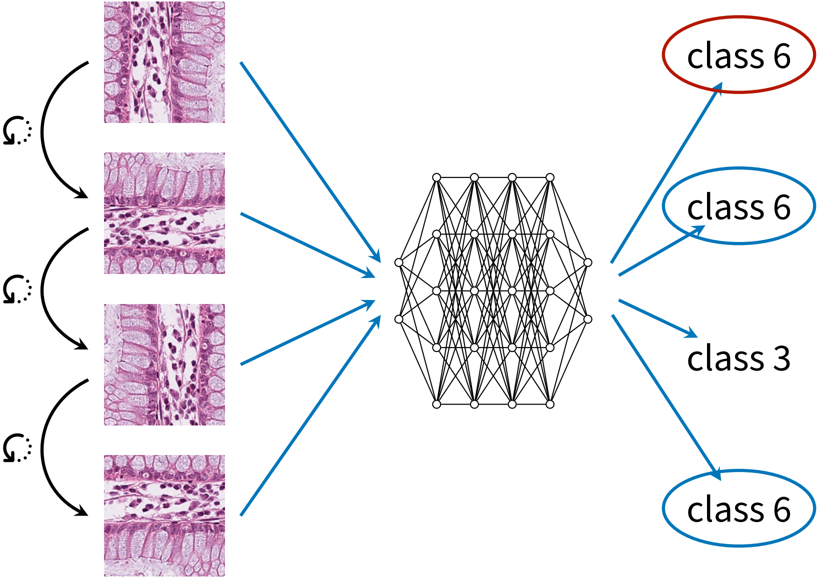
class 6

class 6

class 3

class 6

Histological slices



class 6

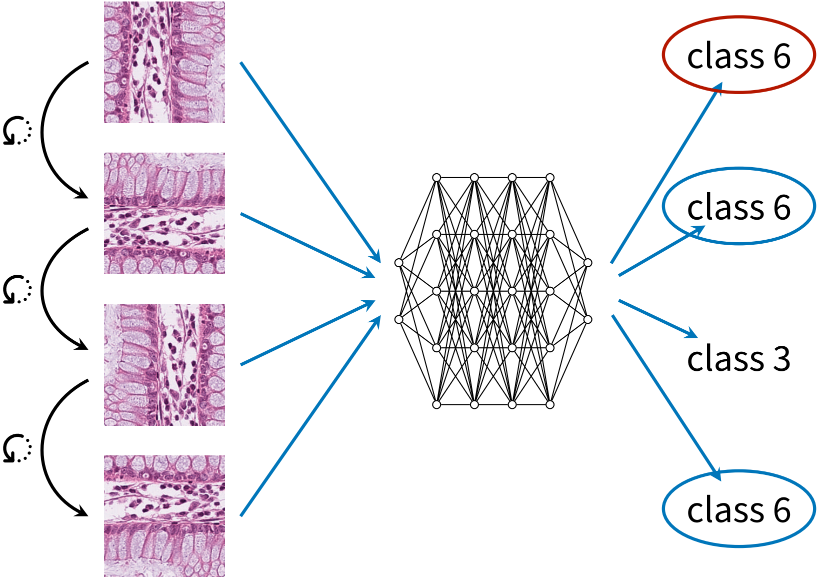
class 6

class 3

class 6

Histological slices

Orbit Same Predictions = 3



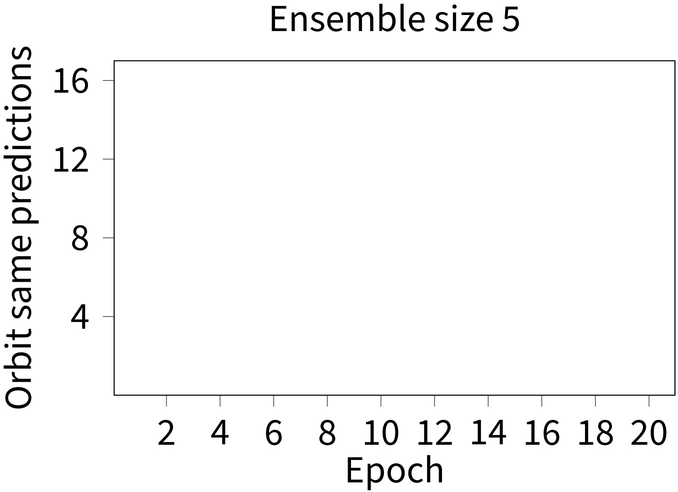
class 6

class 6

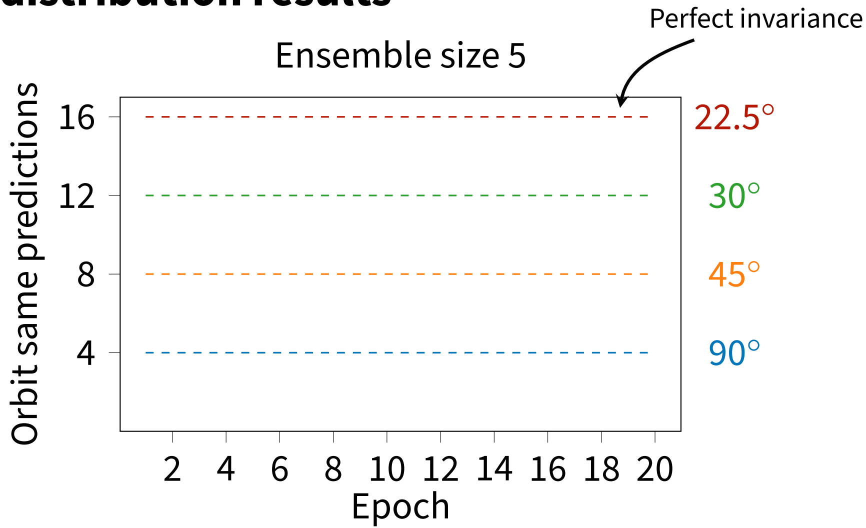
class 3

class 6

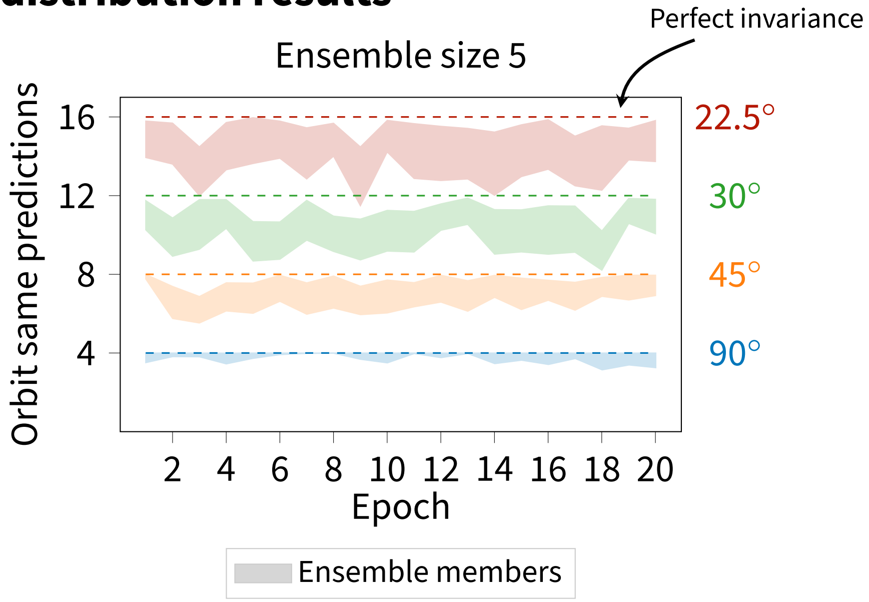
Out of distribution results



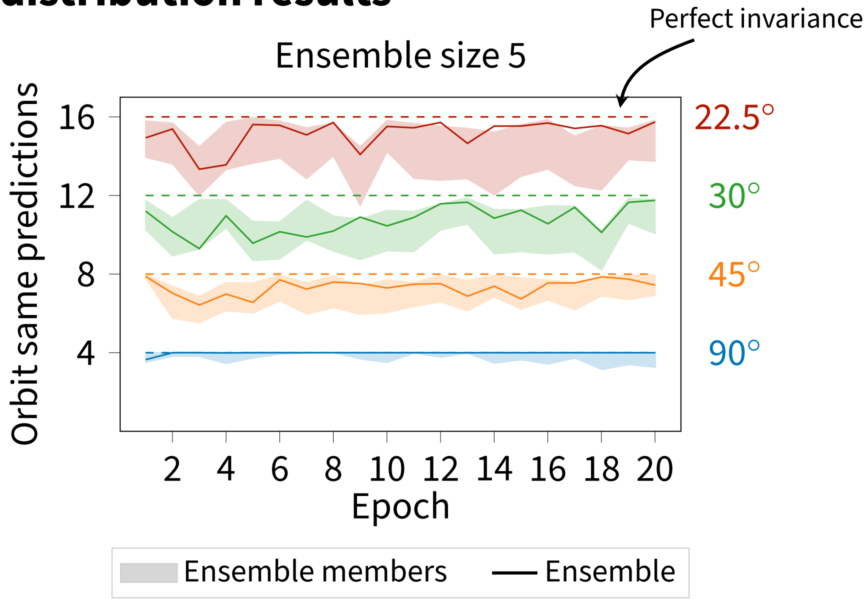
Out of distribution results



Out of distribution results



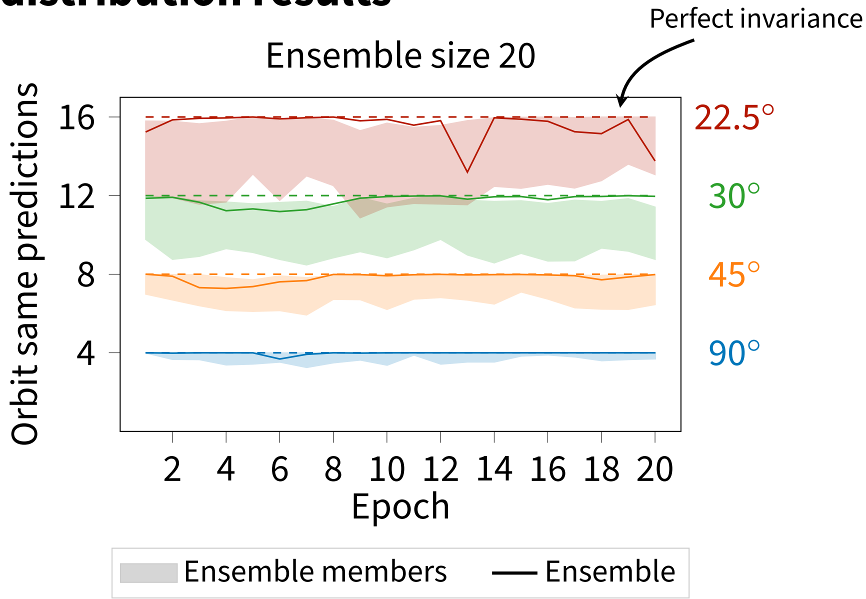
Out of distribution results



Perfect invariance

Ensemble size 5

Out of distribution results



Further experimental results

Further experimental results

✓ Emergent invariance for rotated FashionMNIST

Further experimental results

- ✓ Emergent invariance for rotated FashionMNIST
- ✓ Partial augmentation for continuous symmetries

Further experimental results

- ✓ Emergent invariance for rotated FashionMNIST
- ✓ Partial augmentation for continuous symmetries
- ✓ Emergent equivariance (as opposed to invariance)

Comparison to other methods

Comparison to other methods

➪ Models trained on rotated FashionMNIST

Comparison to other methods

⇨ Models trained on rotated FashionMNIST

Orbit same predictions out of distribution:

	C_4	C_8	C_{16}
DeepEns+DA	3.85±0.12	**7.72±0.34**	**15.24±0.69**
only DA	3.41±0.18	6.73±0.24	12.77±0.71
E2CNN[1]	**4±0.0**	**7.71±0.21**	**15.08±0.34**
Canon[2]	**4±0.0**	**7.45±0.14**	12.41±0.85

[1] [Weiler et al. 2019], [2] [Kaba et al. 2022]

Key takeaways

Key takeaways

If you need ensembles

👍 use data augmentation to obtain an equivariant model.

Key takeaways

If you need ensembles

👍 use data augmentation to obtain an equivariant model.

If you need data augmentation

👍 use an ensemble to boost the equivariance.

Key takeaways

If you need ensembles

👍 use data augmentation to obtain an equivariant model.

If you need data augmentation

👍 use an ensemble to boost the equivariance.

Analysis of neural tangent kernel can lead to powerful practical insights!

Papers

- Geometric deep learning and equivariant neural networks
  Jan E. Gerken, Jimmy Aronsson, Oscar Carlsson, Hampus Linander, Fredrik Ohlsson, Christoffer Petersson, Daniel Persson

  Artificial Intelligence Review 2023
- Emergent Equivariance in Deep Ensembles
  Jan E. Gerken[*], Pan Kessel[*]

  ICML 2024 (Oral)

[*] Equal contribution

Group Website