

Neural Tangent Kernels: Data augmentation and Feynman diagrams

Jan E. Gerken



UNIVERSITY OF
GOTHENBURG



in collaboration with



Pan Kessel



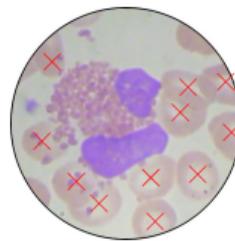
Philipp Misof



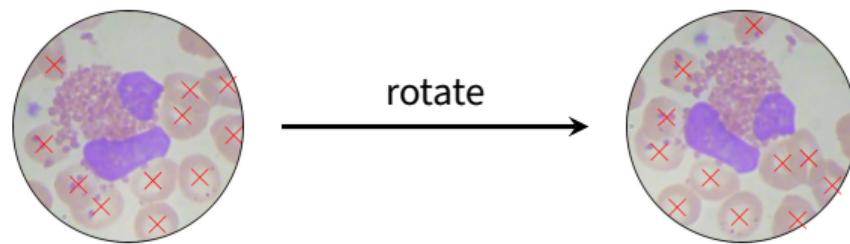
Max Guillen

Symmetries in deep learning

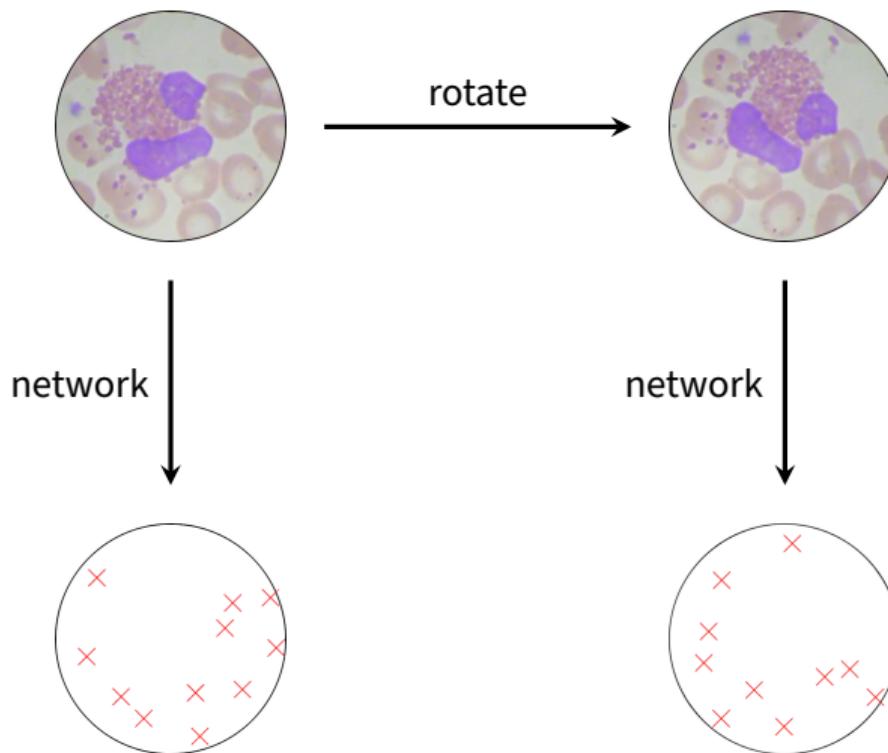
Symmetries in deep learning



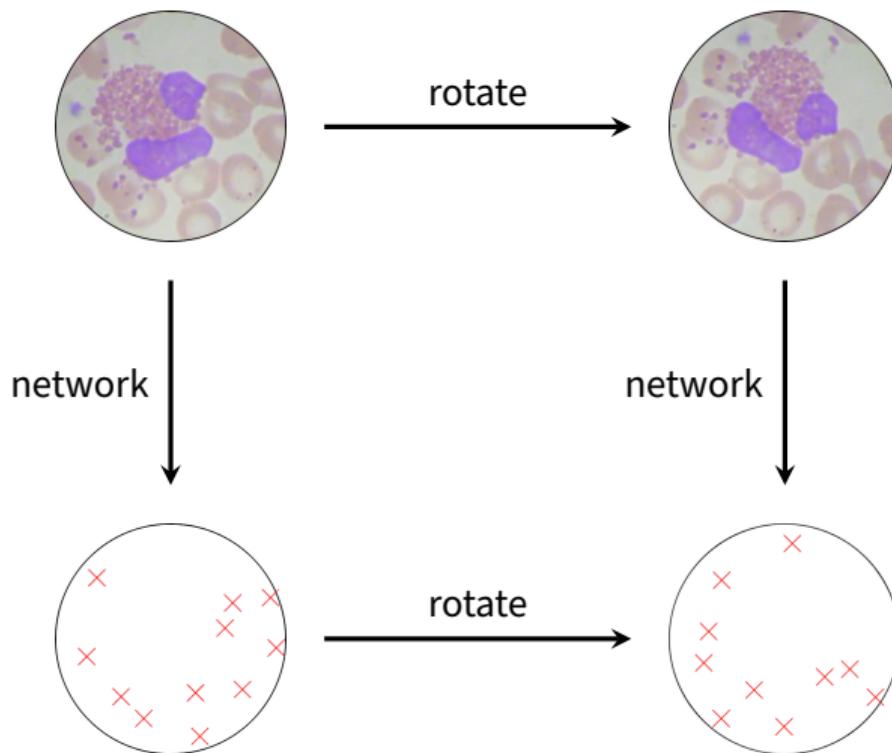
Symmetries in deep learning



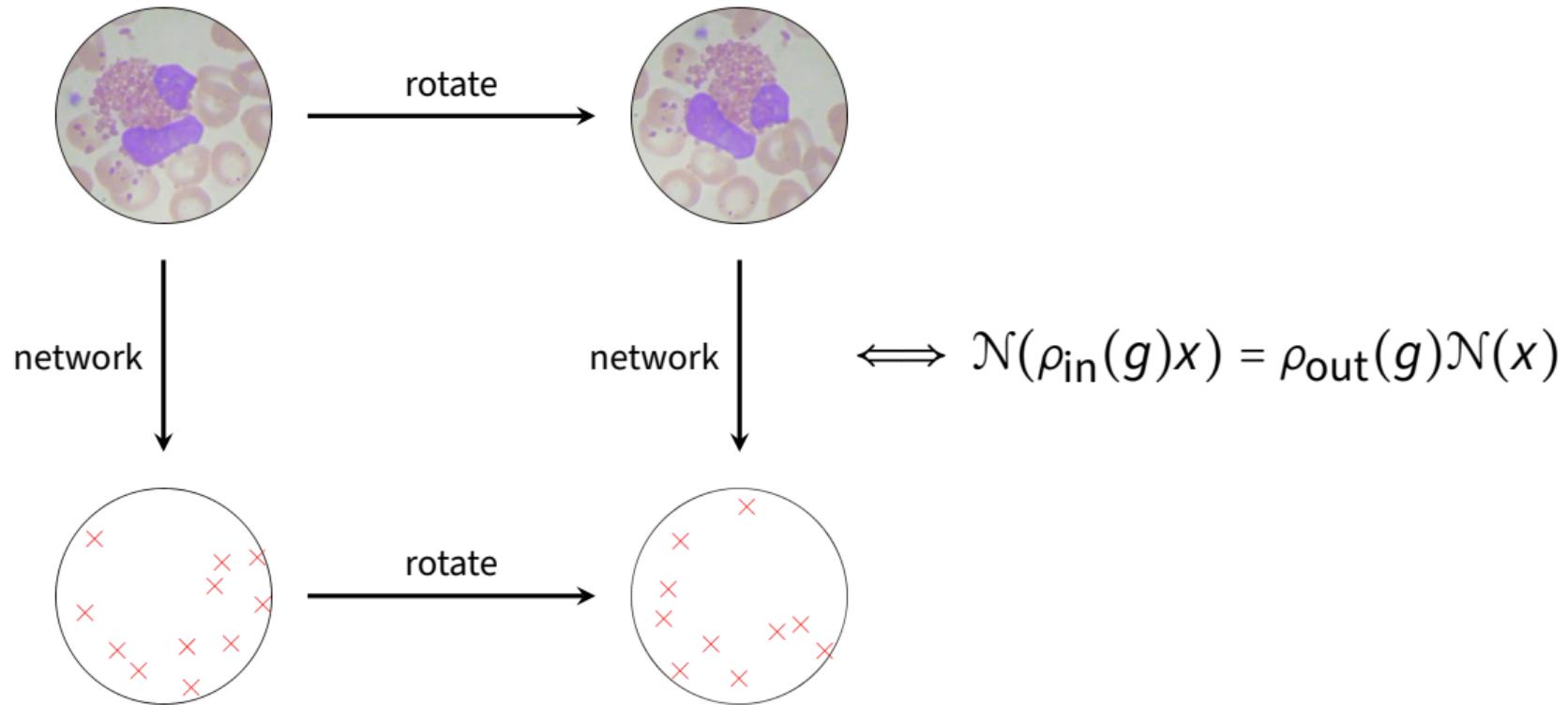
Symmetries in deep learning



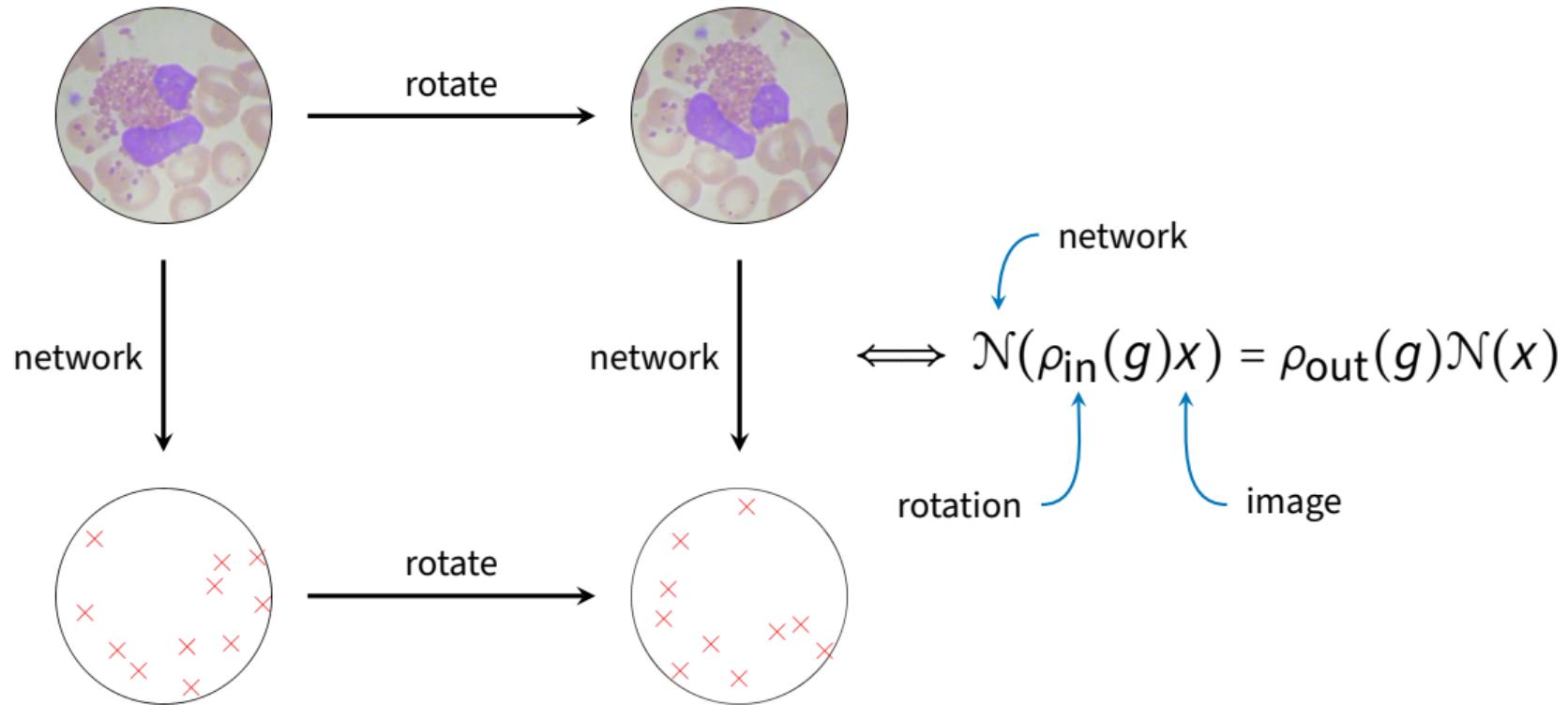
Symmetries in deep learning



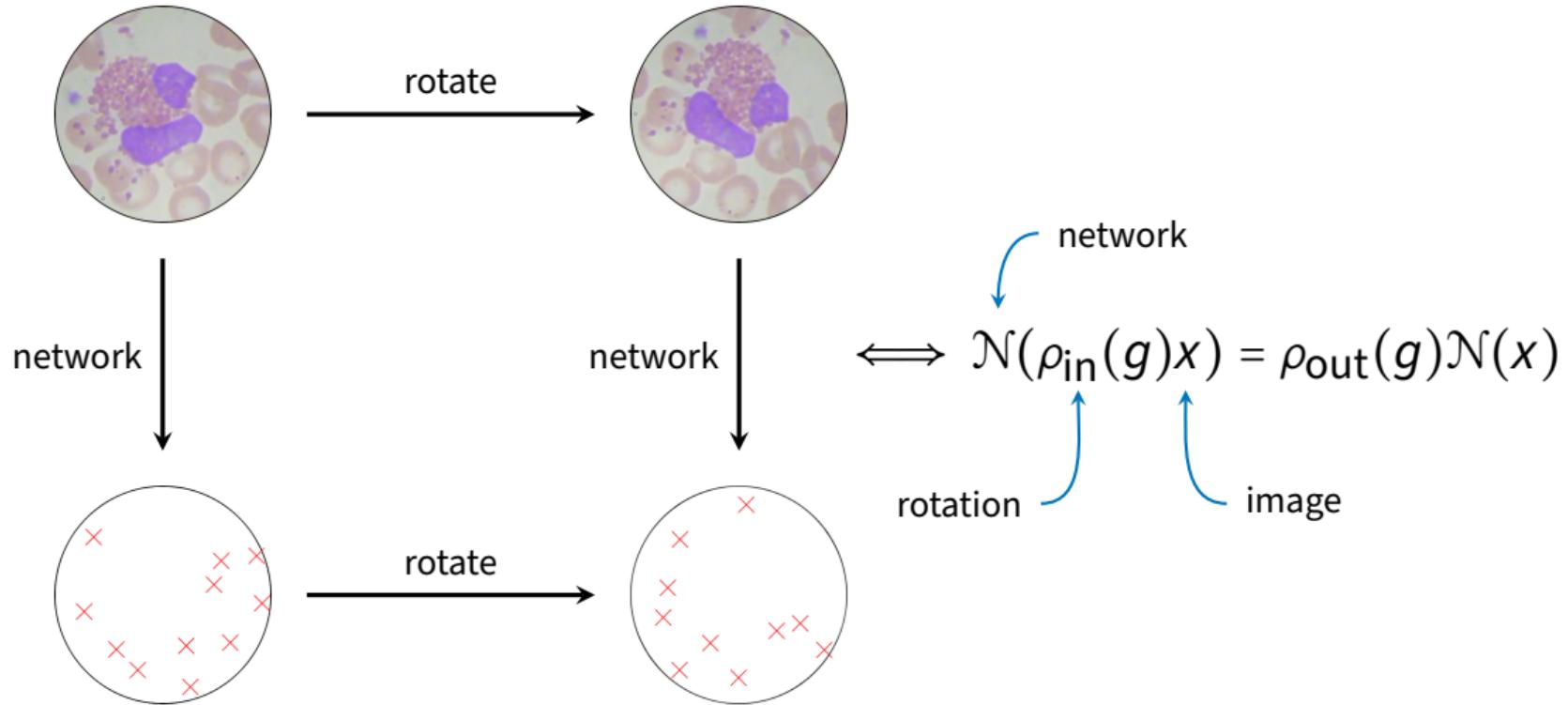
Symmetries in deep learning



Symmetries in deep learning



Equivariance



Equivariant neural networks

Equivariant neural networks

Group Equivariant Convolutional Networks

Taco S. Cohen
University of Amsterdam

T.S.COHEN@UVA.NL

Max Welling
University of Amsterdam
University of California Irvine
Canadian Institute for Advanced Research

M.WELLING@UVA.NL

Abstract

We introduce Group equivariant Convolutional Neural Networks (G-CNNs), a natural generalization of convolutional neural networks that reduces sample complexity by exploiting symme-

Convolution layers can be used effectively in a deep network because all the layers in such a network are *translation equivariant*: shifting the image and then feeding it through a number of layers is the same as feeding the original image through the same layers and then shifting the resulting feature maps (at least up to edge-effects). In

Equivariant neural networks

Group Equivariant Convolutional Networks

Taco S. Cohen

University of Amsterdam

Max Welling

University of Amsterdam

University of California Irvine

Canadian Institute for Advanced Research

T.S.COHEN@UVA.NL

M.WELLING@UVA.NL

Abstract

We introduce Group equivariant Convolutional Neural Networks (G-CNNs), a natural generalization of convolutional neural networks that reduces sample complexity by exploiting symme-

Convolution layers can be used effectively in a deep network because all the layers in such a network are *translation equivariant*: shifting the image and the feeding it through a number of layers is the same as feeding the original image through the same layers and then shifting the resulting feature maps (at least up to edge-effects). In

Equivariant Transformer Networks

Kai Sheng Tai¹ Peter Bailis¹ Gregory Valiant¹

Abstract

How can prior knowledge on the transformation invariance of a domain be incorporated into the architecture of a neural network? We propose Equivariant Transformers (ETs), a family of differentiable function-to-image mappings that preserve the robustness of neural networks to pre-defined continuous transformation groups. Through the use of specially-derived canonical coordinate systems, ETs incorporate functions that

scaling to each training image). While data augmentation typically helps reduce the test error of CNN-based models, there is no guarantee that transformation invariance will be enforced for data not seen during training.

In contrast to training time approaches like data augmentation, recent work on group equivariant CNNs (Cohen & Welling, 2016; Dierkes et al., 2016; Marcus et al., 2017; Woern et al., 2017; Tancik et al., 2017; Alabd, 2017; Cohen et al., 2018) has explored new CNN architectures that are *invariant* to various (nearly-to-be) invariant transformations.

Equivariant neural networks

Group Equivariant Convolutional Networks

Taco S. Cohen

University of Amsterdam

Max Welling

University of Amsterdam

University of California Irvine

Canadian Institute for Advanced Research

T.S.COHEN@UVA.NL

M.WELLING@UVA.NL

Abstract

We introduce Group equivariant Convolutional Neural Networks (G-CNNs), a natural generalization of convolutional neural networks that reduces sample complexity by exploiting symme-

Convolution layers can be used effectively in a deep network because all the layers in such a network are *translation equivariant*: shifting the image and the feeding it through a number of layers is the same as feeding the original image through the same layers and then shifting the resulting feature maps (at least up to edge-effects). In

Equivariant Transformer Networks

Kai Sheng Tai¹ Peter Bailis¹ Gregory Valiant¹

Abstract

How can prior knowledge on the transformation invariance of a domain be incorporated into the architecture of a neural network? We propose Equivariant Transformers (ETs), a family of differentiable end-to-image mappings that preserve the robustness of neural networks to pre-defined continuous transformation groups. Through the use of specially-derived canonical coordinate systems, ETs implement functions that

scaling to each training image). While data augmentation typically helps reduce the test error of CNN-based models, there is no guarantee that transformation invariance will be enforced for data not seen during training.

In contrast to training time approaches like data augmentation, recent work on group equivariant CNNs (Cohen & Welling, 2016; Diefenbach et al., 2016; Marcus et al., 2017; Worrn et al., 2017; Mallya et al., 2017; Mallya, 2017; Cohen et al., 2018) has explored new CNN architectures that are designed to encode modifiable invariance constraints

Theory for Equivariant Quantum Neural Networks

Quynh T. Nguyen,^{1,2} Louis Schatzki,^{3,4} Paolo Branca,^{1,5} Michael Rapone,^{1,6} Patrick J. Coles,³ Frédéric Sauvage,⁴ Martin Lachica,^{1,7} and M. Cirne,³

¹Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA
²School of Engineering and Applied Sciences, Harvard University, Cambridge, Massachusetts 02138, USA

³Information Sciences, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

⁴Department of Electrical and Computer Engineering, University of Illinois Urbana-Champaign, Urbana, Illinois 61801, USA

⁵Department of Mathematics, Aerospace and Mechanical Engineering, University of Southern California, Los Angeles, California 90089-0187, USA

⁶Department of Mathematics, University of California Davis, Davis, California 95616, USA
⁷Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

Quantum neural network architectures that have little-to-no inductive biases are known to face trainability and generalization issues. Inspired by a similar problem, recent breakthroughs in machine learning address this challenge by creating models encoding the symmetries of the learning task. This is materialized through the usage of equivariant neural networks whose action commutes with that of the symmetry. In this work we extend these ideas to the quantum regime by

Equivariant neural networks

Group Equivariant Convolutional Networks

Taco S. Cohen

University of Amsterdam

Max Welling

University of Amsterdam

University of California Irvine

Canadian Institute for Advanced Research

T.S.COHEN@UVA.NL

M.WELLING@UVA.NL

Abstract

We introduce Group equivariant Convolutional Neural Networks (G-CNNs), a natural generalization of convolutional neural networks that reduces sample complexity by exploiting symme-

Convolution layers can be used effectively in a deep network because all the layers in such a network are *translation equivariant*: shifting the image and feeding it through a number of layers is the same as feeding the original image through the same layers and then shifting the resulting feature maps (at least up to edge-effects). In

Equivariant Transformer Networks

Kai Sheng Tai¹ Peter Bailis¹ Gregory Valiant¹

Abstract

How can prior knowledge on the transformation invariance of a domain be incorporated into the architecture of a neural network? We propose Equivariant Transformers (ETs), a family of differentiable end-to-image mappings that preserve the robustness of neural networks to pre-defined continuous transformation groups. Through the use of specially-derived canonical coordinate systems, ETs implement functions that

scaling to each training image). While data augmentation typically helps reduce the test error of CNN-based models, there is no guarantee that transformation invariance will be enforced for data not seen during training.

In contrast to training time approaches like data augmentation, recent work on group equivariant CNNs (Cohen & Welling, 2016; Diefenbach et al., 2016; Marcus et al., 2017; Worrall et al., 2017; Mallya et al., 2017; Cohen et al., 2018) has explored new CNN architectures that are designed to encode modifiable invariance constraints.

Theory for Equivariant Quantum Neural Networks

Quynh T. Nguyen,^{1,2} Louis Schatzki,^{3,4} Paolo Branca,^{1,5} Michael Rapone,^{1,6} Patrick J. Coles,³ Frédéric Sauvage,⁴ Martin Laecca,^{1,7} and M. Cirone³

¹Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA
²School of Engineering and Applied Sciences, Harvard University, Cambridge, Massachusetts 02138, USA

³Information Sciences, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

⁴Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA

⁵Department of Physics, Aerospace and Mechanical Engineering, University of Southern California, Los Angeles, California 90089, USA

⁶Department of Mathematics, University of California Davis, Davis, California 95616, USA

⁷Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

Quantum neural network architectures that have little-to-no inductive biases are known to face trainability and generalization issues. Inspired by a similar problem, recent breakthroughs in machine learning address this challenge by creating models encoding the symmetries of the learning task. This is materialized through the usage of equivariant neural networks whose action commutes with that of the symmetry. In this work we extend these ideas to the quantum setting by

An Efficient Lorentz Equivariant Graph Neural Network for Jet Tagging

Shiqi Gong^{a,c,1} Qi Meng^b Jue Zhang^b Huilin Qu^c Congqiao Li^c Sitian Qian^d Weitao Du^a Zhi-Ming Ma^a Tie-Yan Liu^b

^aAcademy of Mathematics and Systems Science, Chinese Academy of Sciences,
Zhongguancun East Road, Beijing 100190, China

^bMicrosoft Research Asia,
Duning Street, Beijing 100089, China

^cCERN, EP Department,
CH-1211 Geneva 23, Switzerland

^dSchool of Physics, Peking University,
Chenfu Road, Beijing 100871, China

Equivariant neural networks

Group Equivariant Convolutional Networks

Taco S. Cohen

University of Amsterdam

Max Welling

University of Amsterdam

University of California Irvine

Canadian Institute for Advanced Research

T.S.COHEN@UVA.NL

M.WELLING@UVA.NL

Abstract

We introduce Group equivariant Convolutional Neural Networks (G-CNNs), a natural generalization of convolutional neural networks that reduces sample complexity by exploiting symme-

Convolution layers can be used effectively in a deep network because all the layers in such a network are *translation equivariant*: shifting the image and feeding it through a number of layers is the same as feeding the original image through the same layers and then shifting the resulting feature maps (at least up to edge-effects). In

An Efficient Lorentz Equivariant Graph Neural Network for Jet Tagging

Shiqi Gong^{a,1}, Qi Meng^b, Jue Zhang^b, Huilin Qu^c, Congqiao Li^c, Sitian Qian^d, Weitao Du^a, Zhi-Ming Ma^a, Tie-Yan Liu^b

^aAcademy of Mathematics and Systems Science, Chinese Academy of Sciences,
Zhongguancun East Road, Beijing 100190, China

^bMicrosoft Research Asia,
Daming Street, Beijing 100089, China

^cCERN, EP Department,
CH-1211 Geneva 23, Switzerland

^dSchool of Physics, Peking University,
Chenfu Road, Beijing 100871, China

Equivariant Transformer Networks

Kai Sheng Tai¹, Peter Bailis¹, Gregory Valiant¹

Abstract

How can prior knowledge on the transformation invariance of a domain be incorporated into the architecture of a neural network? We propose Equivariant Transformers (ETs), a family of differentiable end-to-image mappings that preserve the robustness of neural networks to pre-defined continuous transformation groups. Through the use of symmetry-derived canonical coordinate systems, ETs incorporate functions that

scaling to each training image). While data augmentation typically helps reduce the test error of CNN-based models, there is no guarantee that transformation invariance will be enforced for data not seen during training.

In contrast to training time approaches like data augmentation, recent work on group equivariant CNNs (Cohen & Welling, 2016; Diefenbach et al., 2016; Marcus et al., 2017; Worrall et al., 2017; Mallya et al., 2017; Cohen et al., 2018) has explored new CNN architectures that are

Theory for Equivariant Quantum Neural Networks

Quynh T. Nguyen,^{1,2} Louis Schatzki,^{3,4} Paolo Branca,^{1,5} Michael Rapone,^{1,6}
Patrick J. Coles,³ Frédéric Sauvage,⁴ Martin Laecca,^{1,7} and M. Cirone³

¹Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA
²School of Engineering and Applied Sciences, Harvard University, Cambridge, Massachusetts 02138, USA

³Information Sciences, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

⁴Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA

⁵Department of Physics, Aerospace and Mechanical Engineering, Santa Fe Institute, Santa Fe, New Mexico 87501, USA

⁶Department of Mathematics, University of California Davis, Davis, California 95616, USA

⁷Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

Quantum neural network architectures that have little-to-no inductive biases are known to face quantum stability and generalization issues. Inspired by a similar problem, recent breakthroughs in machine learning address this challenge by creating models encoding the symmetries of the learning task. This is materialized through the usage of equivariant neural networks whose action commutes with that of the symmetry. In this work we present theory for the quantum analogues for

E(3)-Equivariant Graph Neural Networks for Data-Efficient and Accurate Interatomic Potentials

Simon Batzner^{a,1}, Albert Musoelian¹, Lixin Sun¹, Mario Geiger², Jonathan P. Mallon³,
Mordechai Kornblith², Nicola Molinari¹, Tess E. Smith^{4,5} and Boris Kozinsky^{a,3}

¹John A. Paulson School of Engineering and Applied Sciences,
Harvard University, Cambridge, MA 02138, USA

²École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland
³Robert Bosch Research and Technology Center, Cambridge, MA 02120, USA

⁴Computational Research Division and Center for Advanced Computing for Energy Research Applications,

Lawrence Berkeley National Laboratory, Berkeley, CA 94730, USA

⁵Massachusetts Institute of Technology, Department of Electrical
Engineering and Computer Science, Cambridge, MA 02139, USA

This work presents Neural Equivariant Interatomic Potentials (NeqIP), an E(3)-equivariant neural network approach for learning interatomic potentials from *ab-initio* calculations for molecular dynamics simulations. While most contemporary symmetry-aware models use invariant convolutions and only act on scalars, NeqIP employs E(3)-equivariant convolutions for interactions of geometric tensors, resulting in a more information-rich and faithful representation of atomic environments. The method achieves state-of-the-art accuracy on a challenging and diverse set of molecules and

Equivariant neural networks

Group Equivariant Convolutional Networks

Taco S. Cohen

University of Amsterdam

Max Welling

University of Amsterdam

University of California Irvine

Canadian Institute for Advanced Research

T.S.COHEN@UVA.NL

M.WELLING@UVA.NL

Abstract

We introduce Group equivariant Convolutional Neural Networks (G-CNNs), a natural generalization of convolutional neural networks that reduces sample complexity by exploiting symme-

Convolution layers can be used effectively in a deep network because all the layers in such a network are *translation equivariant*: shifting the image and feeding it through a number of layers is the same as feeding the original image through the same layers and then shifting the resulting feature maps (at least up to edge-effects). In

Equivariant Transformer Networks

Kai Sheng Tai¹ Peter Bailis¹ Gregory Valiant¹

Abstract

How can prior knowledge on the transformation invariance of a domain be incorporated into the architecture of a neural network? We propose Equivariant Transformers (ETs), a family of differentiable end-to-image mappings that preserve the robustness of neural networks to pre-defined continuous transformations. Through the use of specially-derived canonical coordinate systems, ETs implement functions that

scaling to each training image). While data augmentation typically helps reduce the test error of CNN-based models, there is no guarantee that transformation invariance will be enforced for data not seen during training.

In contrast to training time approaches like data augmentation, recent work on group equivariant CNNs (Cohen & Welling, 2016; Dilemmas et al., 2016; Marcus et al., 2017; Worrn et al., 2017; Mallya et al., 2017; Cohen et al., 2018) has explored new CNN architectures that are designed to encode modularity via invariant mechanisms

Theory for Equivariant Quantum Neural Networks

Quynh T. Nguyen,^{1,2} Louis Schatzki,^{3,4} Paolo Branca,^{1,5} Michael Rapone,^{1,6} Patrick J. Coles,¹ Frédéric Sauvage,⁴ Martin Laločka,^{1,7} and M. Černý³

¹Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA
²School of Engineering and Applied Sciences, Harvard University, Cambridge, Massachusetts 02138, USA

³Information Sciences, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

⁴Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA

⁵Department of Mathematics, University of Southern California, Los Angeles, California 90089, USA

⁶Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA

Quantum neural network architectures that have little-to-no inductive biases are known to face trainability and generalization issues. Inspired by a similar problem, recent breakthroughs in machine learning address this challenge by creating models encoding the symmetries of the learning task. This is materialized through the usage of equivariant neural networks whose action commutes with that of the symmetry. In this work we present those to the quantum setting

HIERARCHICAL, ROTATION-EQUIVARIANT NEURAL NETWORKS TO SELECT STRUCTURAL MODELS OF PROTEIN COMPLEXES

Stephan Eismann*

Department of Applied Physics
Stanford University
seismann@stanford.edu

Raphael J.L. Townsend*

Department of Computer Science
Stanford University
raphael@cs.stanford.edu

Nathaniel Thomas*

Department of Physics
Stanford University
nthomas103@gmail.com

Mihail Jagota

Department of Electrical Engineering
Stanford University
mjagota@stanford.edu

Bowen Jing

Department of Computer Science
Stanford University
bjing@cs.stanford.edu

Ron O. Dror

Department of Computer Science
Stanford University
rondror@cs.stanford.edu

ABSTRACT

Predicting the structure of multi-protein complexes is a grand challenge in biochemistry, with major implications for basic science and drug discovery. Computational structure prediction methods generally leverage pre-defined structural features to distinguish accurate structural models from less accurate ones. This raises the question of whether it is possible to learn characteristics of accurate models directly from atomic coordinates of protein complexes, with no prior assumptions. Here we introduce a machine learning method that learns directly from the 3D positions of all atoms to

An Efficient Lorentz Equivariant Graph Neural Network for Jet Tagging

Shiqi Gong^{a,1} Qi Meng^b Jue Zhang^b Huilin Qu^c Congqiao Li^c Sitian Qian^d Weitao Du^a Zhi-Ming Ma^a Tie-Yan Liu^b

^aAcademy of Mathematics and Systems Science, Chinese Academy of Sciences, Zhongguancun East Road, Beijing 100190, China

^bMicrosoft Research Asia, Daming Street, Beijing 100089, China

^cCERN, EP Department, CH-1211 Geneva 23, Switzerland

^dSchool of Physics, Peking University, Chenfu Road, Beijing 100871, China

E(3)-Equivariant Graph Neural Networks for Data-Efficient and Accurate Interatomic Potentials

Simon Batzner^{a,1} Albert Musoelian¹ Lixin Sun¹ Mario Geiger² Jonathan P. Mallon³ Mordechai Kornbluth² Nicola Molinari¹ Tess E. Smith^{4,5} and Boris Kozinsky^{a,1,3}

¹John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138, USA

²École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland

³Robert Bosch Research and Technology Center, Cambridge, MA 02120, USA

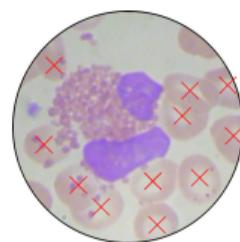
⁴Computational Research Division, and Center for Advanced Computing for Energy Research Applications, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

⁵Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, Cambridge, MA 02132, USA

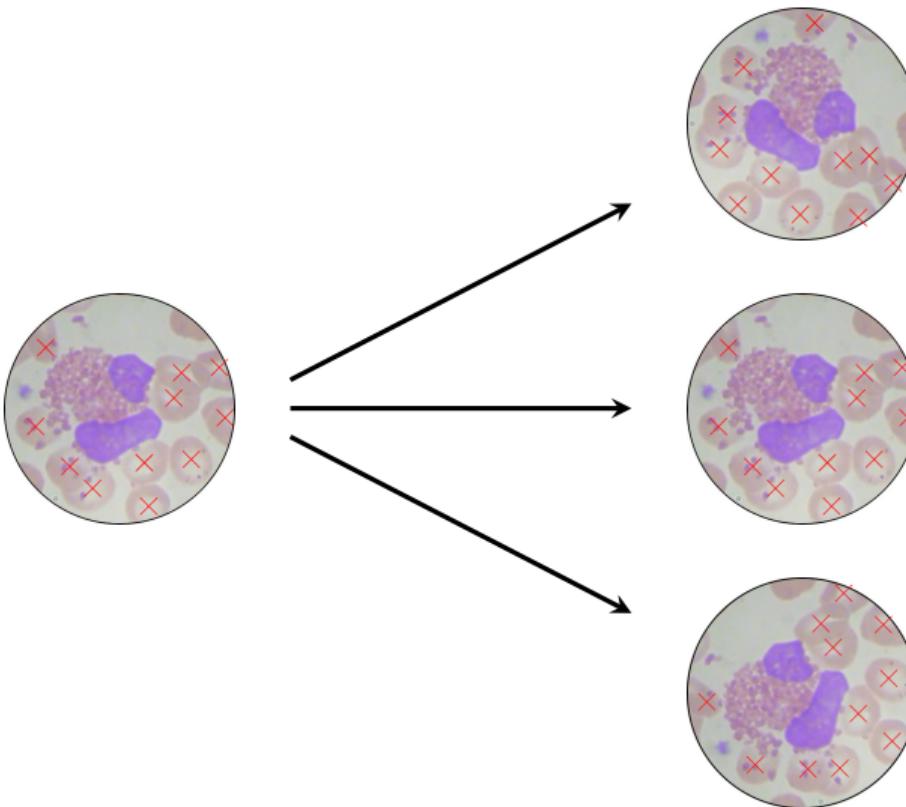
This work presents Neural Equivariant Interatomic Potentials (NeqIP), an E(3)-equivariant neural network approach for learning interatomic potentials from *ab-initio* calculations for molecular dynamics simulations. While most contemporary symmetry-aware models use invariant convolutions and only act on scalars, NeqIP employs E(3)-equivariant convolutions for interactions of geometric tensors, resulting in a more information-rich and faithful representation of atomic environments. The method achieves state-of-the-art accuracy on a challenging and diverse set of molecules and

Data augmentation

Data augmentation



Data augmentation



Data augmentation

Data augmentation

Article

Accurate structure prediction of biomolecular interactions with AlphaFold 3

<https://doi.org/10.1038/s41589-024-07487-w>

Received: 19 December 2023

Accepted: 29 April 2024

Published online: 8 May 2024

Open access

Check for updates

Josh Abramson^{1,2}, Jonas Adler³, Jack Danger^{1,4}, Richard Evans⁵, Tim Green^{2,6}, Alexander Pritzel^{2,7}, Olaf Ronneberger⁸, Lindsay Williams⁹, Andrew J. Bellard¹⁰, Jennifer C. Boudelle¹¹, Sebastian V. Bräuer¹², David A. Brown¹³, Chaitanya Chakraborty¹⁴, Michael O’Neill¹⁵, Daniel Riedel¹⁶, Kathryn Sawyer¹⁷, Zuchan Wang¹⁸, Nalini Sengupta¹⁹, Estri Aravant²⁰, Charles Boettig²¹, Ottavia Bartoli²², Alex Bridgland²³, Aleney Chempurac²⁴, Miles Congreve²⁵, Alexander L. Cowen-Rivers²⁶, Andrew Cowie²⁷, Michael Figariros²⁸, Michael Gromiha²⁹, Michael Dauter³⁰, Michael E. Groll³¹, Yannick K. Hildebrand³², Catherine M. H. Lew³³, Koko Peris³⁴, Anil Pratap³⁵, Pouyat Rave³⁶, Sulabh Ray³⁷, Adrien Reymond³⁸, Ashvika Thivikosambaran³⁹, Catherine Tong⁴⁰, Sergei Vakser⁴¹, Ellen S. Zhang⁴², Michal Zelma⁴³, Augustin Žíšek⁴⁴, Victor Kapit⁴⁵, Pudvezet Kath⁴⁶, Max Jaderberg^{47,48}, Dennis Hesselbar^{49,50} & John M. Jumper^{1,2,51}

The introduction of AlphaFold 2 has spurred a revolution in modelling the structure of proteins and their interactions, enabling a huge range of applications in protein modelling and design^{1–4}. Here we describe the AlphaFold 3 model with a substantially updated architecture that can predict the structures of proteins and complexes of complex systems including proteins, nucleic acids, small molecules, ions and modified residues. The new AlphaFold model demonstrates substantially improved accuracy

Data augmentation

Article

Accurate structure prediction of biomolecular interactions with AlphaFold 3

<https://doi.org/10.1038/s41589-024-07487-w>

Received: 19 December 2023

Accepted: 29 April 2024

Published online: 8 May 2024

Open access

 Check for updates

Josh Abramson^{1,2}, Alexander Pritzel^{1,2}, Richard Evans³, Tim Green^{2,4}, Tony Willmetts⁵, Andrew J. Ballard⁶, Michael O'Neil⁷, David A. Baker⁸, Chaitin Hwang⁹, Daniel Hartmann¹⁰, Zoltan Szabadits¹¹, Sambasiva Rengaraju¹², Eva Bartels¹³, Alex Bridgland¹⁴, Alenay Chempuranc¹⁵, Miles Caren¹⁶, Daniel Chen¹⁷, Yannick Chêne¹⁸, Catherine H. R. Lew¹⁹, Paul F. D. Flory²⁰, Sulabh Goyal²¹, Adrien Henrion²², Catherine Tong²³, Sergei Vakser²⁴, Ellen S. Zhang²⁵, Martin Židák²⁶, Victor Kapur²⁷, Pudvezet Kachl²⁸, Max Jaderberg^{29,30}, & John M. Jumper^{1,2,31}

The introduction of AlphaFold 2 has spurred a revolution in modelling the structure of proteins and their interactions, enabling a huge range of applications in protein modelling and design^{1–3}. Here we describe the AlphaFold 3 model with a substantially updated set of biomolecular interactions, including predictions of the structures of complex mixtures including proteins, nucleic acids, small molecules, ions and modified residues. The new AlphaFold model demonstrates substantially improved accuracy

No Equivariance

Data augmentation

Article

Accurate structure prediction of biomolecular interactions by AlphaFold 3

<https://doi.org/10.1038/s4324-024-07487-w>

Received: 19 December 2023

Accepted: 29 April 2024

Published online: 8 May 2024

Open access

 Check for updates

No Equivariance¹

Josh Abramson^{1,2}, Alexander Pritzel^{1,2}, Richard Evans³, Tim Green^{2,4}, Tony Willmetts², Andrew J. Ballard², Michael O'Neil², David A. Kuhns², Chaitanya Raghav², Michael O'Neil², Harvey M. Rosenblatt², Zachariah S. Biegel², Divya Bartsch², Alex Bridgland², Alenay Chempak², Miles Cai², Daniel Cai², Daniel Young², Kyoko Yamada², Christopher H. R. Lew², Catherine Tong², Sergei Vakser², Ellen S. Zhang², Martin Zídek², Victor Kapur², Pudvezan Kathi², Max Jaderberg^{2,5}, & John M. Jumper^{2,6}

The introduction of AlphaFold 2 has spurred a revolution in modelling the structure of proteins and their interactions, enabling a huge range of applications in protein modelling and design^{1–3}. Here we describe the AlphaFold 3 model with a substantially updated set of biomolecular interaction predictions, capable of predicting the structure of complex systems including proteins, nucleic acids, small molecules, ions and modified residues. The new AlphaFold model demonstrates substantially improved accuracy

The Importance of Being Scalable: Improving the Speed and Accuracy of Neural Network Interatomic Potentials Across Chemical Domains

Eric Qu¹
UC Berkeley
ericqu@berkeley.edu

Aditi S. Krishnapriyan¹
UC Berkeley, LBNL
aditik1@berkeley.edu

Abstract

Scaling has been a critical factor in improving model performance and generalization across various fields of machine learning. It involves how a model's performance changes with increases in model size or input data, as well as how efficiently computational resources are utilized to support this growth. Despite successes in scaling other types of machine learning models, the study of scaling in Neural Network Interatomic Potentials (NNIPs) remains limited. NNIPs act as

Data augmentation

Article

Accurate structure prediction of biomolecular interactions with AlphaFold 3

<https://doi.org/10.1038/s4324-024-07487-w>

Received: 19 December 2023

Accepted: 29 April 2024

Published online: 8 May 2024

Open access

Check for updates

No Equivariance

The introduction of AlphaFold 2 has spurred a revolution in modelling the structure of proteins and their interactions, enabling a huge range of applications in protein modelling and design^{1–3}. Here we describe the AlphaFold 3 model with a substantially updated architecture that achieves state-of-the-art performance across a wide range of complex systems including proteins, nucleic acids, small molecules, ions and modified residues. The new AlphaFold model demonstrates substantially improved accuracy

Josh Abrahamsen^{1,2}, Alexander Pritzel^{1,2}, Michael O’Hearn^{1,2}, Richard Evans^{1,2}, Tim Green^{1,2}, Alexander Prizel^{1,2}, Michael O’Hearn^{1,2}, Richard Evans^{1,2}, Tim Green^{1,2}, Andrew J. Ballard^{1,2}, Tony Willmott^{1,2}, Andrew J. Ballard^{1,2}, Tony Willmott^{1,2}, Michael O’Hearn^{1,2}, Richard Evans^{1,2}, Tim Green^{1,2}, David A. Kortemme^{1,2}, Chaitanya R. Gajalakshmi^{1,2}, Zohreh Ebrahimi^{1,2}, Sambasiva Rengaraju^{1,2}, Divya Bartsch^{1,2}, Alex Brigandt^{1,2}, Alenay Chempur^{1,2}, Estri Arguello Miles^{1,2}, Fabio Caccamo^{1,2}, Daniel Young^{1,2}, Kieren C. Coombes^{1,2}, M. R. Lew^{1,2}, Mark A. Peacock^{1,2}, Sulabh Singh^{1,2}, Adam Thompson^{1,2}, Catherine Tong^{1,2}, Sergei Vakser^{1,2}, Ellen S. Zhang^{1,2}, Martin Zídek^{1,2}, Victor Kapur^{1,2}, Pudvezan Kathi^{1,2}, Max Jaderberg^{1,2}, & John M. Jumper^{1,2}

The Importance of Being Scalable: Improving the Speed and Accuracy of Neural Network Interatomic Potentials Across Chemical Domains

Eric Qu¹
UC Berkeley
ericqu@berkeley.edu

Aditi S. Krishnapriyan¹
UC Berkeley, LBNL
aditik1@berkeley.edu

Abstract

Scaling has been a critical factor in improving model performance and generalization across various fields of machine learning. It involves how a model’s performance changes with increases in model size or input data, as well as how efficiently computational resources are utilized to support this growth. Despite successes in scaling other types of machine learning models, the study of scaling in Neural Network Interatomic Potentials (NNIPs) remains limited. NNIPs act as

Swallowing the Bitter Pill: Simplified Scalable Conformer Generation

Yuyang Wang¹, Ahmed A. Elbag^{1,2}, Navdeep Jolly¹, Joshua M. Susskind¹, Miguel Ángel Bautista¹

Abstract

We present a novel way to predict molecular conformers through a simple formulation that sidesteps many of the heuristics of prior works and achieves state of the art results by using the advantages of scale. By training a diffusion generative model directly on 3D atomic positions without any constraints about the chemical structure of molecules (or, more precisely, bond angles) we are able to radically simplify structure generation and predict conformers even when

is the vast complexity of the 3D structure space, encompassing factors such as bond lengths and torsional angles. Designing the molecular conformer space with no constraints on dihedral specific constraints, such as bond types and spatial arrangements determined by chiral centers, the conformational space experiences exponential growth with the expansion of the graph size and the number of rotatable bonds (Aszkenasy & Gomez-Bombarelli, 2022). This complicates brute force and exhaustive approaches, making them virtually unusable for even moderately small molecules. Systematic methods, like DMTGA (Hawkins et al., 2018),

Data augmentation

Article

Accurate structure prediction of biomolecular interactions with AlphaFold 3

<https://www.minsport.gov.ru>

[View Details](#)

Received: 19 December 2013

Accepted: 29 April 2024

www.wiley.com/go/robinson/

Published online: 28 Dec

Chinese edition

No Equivariance!

Josh Alemian^{1,2}, Alexander Pritzel¹, Richard Eason³,
Xiangyu Chen¹, Henry Willcocks¹,
Michael O’Neill¹, David A. Rosen¹,
Evan Ackerman¹, Daniel Berthelot¹,
Miles Brundage¹, Andrew Borsig¹,
Pete Cai¹, Michael Burda¹,
Catherine Corgat¹, Benjamin Dyer¹,
Austin Estevez¹, Philipp Hennig¹,
and John M. Jumper^{1,2}

1. Production of AlphaFold 2 has spurred new developments in the field, enabling a huge leap forward in the prediction of protein structures and the interactions between them.

Production of AlphaFold 2 has spurred a revolution in modelling the structure of proteins and their interactions, enabling a huge range of applications in protein modelling and design.^{6–9} Here we describe our AlphaFold 3 model with a substantially updated diffusion-based architecture that is capable of predicting the joint structure of complexes including proteins, nucleic acids, small molecules, ions and modified residues. The new AlphaFold model demonstrates substantially improved accuracy.

Probing the effects of broken symmetries in machine learning

Marcel F Langer¹, Sergey N Pozdnyakov² and Michele Ceriotti^{1,3}

Laboratory of Computational Science and Modeling and National Centre for Computational Design and Discovery of Novel Materials MARVEL, Institute of Materials, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland

* Author to whom any correspondence should be addressed.

E-mail: richard@csail.mit.edu

Keywords: *multilevel regression, community-based training, middle, stochastic modelling, education, circulation*

Journal of Health Politics, Policy and Law, Vol. 33, No. 3, June 2008
DOI 10.1215/03616878-33-3-621 © 2008 by the Southern Political Science Association

4. [About us](#)

Symmetry is one of the most central concepts in physics, and it is no surprise that it has also been widely adopted as an inductive bias for machine-learning models applied to the physical sciences. This is especially true for models targeting the properties of matter at the atomic scale. Both established and state-of-the-art approaches, with almost no exceptions, are built to be exactly equivariant to translations, permutations, and rotations of the atoms. Incorporating symmetries—rotations in particular—constrains the model design space and implies more complicated architectures that are often also computationally demanding. There are indications

The Importance of Being Scalable: Improving the Speed and Accuracy of Neural Network Interatomic Potentials Across Chemical Domains

Eric Qu
UC Berkeley
ericqu@berkeley.edu

Aditi S. Krishnapriyan
UC Berkeley, LBNL
aditik@berkeley.edu

Abstract

Scaling has been a critical factor in improving model performance and generalization across various fields of machine learning. It involves how a model's performance changes with increases in model size or input data, as well as how efficiently computational resources are utilized to support this growth. Despite successes in scaling other types of machine learning models, the study of scaling in Neural Network Interatomic Potentials (NNIPs) remains limited. NNIPs act as

Following the Bitter Pill: Simplified Scalable Conformer Generation

Yuyang Wang¹ Ahmed A. Elbag^{1,2} Navdeep Jaitly³ Joshua M. Susskind⁴ Miguel Ángel Bautista⁵

Abstract

We present a novel way to predict molecular conformations through a simpler formulation that sidesteps many of the heuristics of prior works and achieves state of the art results by using the advantages of scale. By training a diffusion generative model directly on 3D atomic positions without making assumptions about the explicit structure of molecules (e.g. modeling torsional angles) we are able to radically simplify structure learning and enable statistical methods on the

is the vast complexity of the 3D structure space, encompassing factors such as bond lengths and torsional angles. Despite the molecular graph dictating potential 3D conformers through specific constraints, such as bond types and spatial arrangements determined by chiral centers, the conformational space experiences exponential growth with the expansion of the graph size and the number of rotatable bonds (Aszledi & Gomez-Bombarelli, 2022). This complicates brute force and exhaustive approaches, making them virtually unfeasible for even moderately small molecules.

Data augmentation

Article

Accurate structure prediction of biomolecular interactions with AlphaFold 3

<https://doi.org/10.1038/s43998-024-02487-w>

Received: 19 December 2023

Accepted: 29 April 2024

Published online: 8 May 2024

Open access

Check for updates

No Equivariance

The introduction of AlphaFold 2 has spurred a revolution in modelling the structure of proteins and their interactions, enabling a huge range of applications in protein modelling and design^{1–3}. Here we describe the AlphaFold 3 model with substantially updated architecture and improved accuracy of protein structure prediction of complex systems including proteins, nucleic acids, small molecules, ions and modified residues. The new AlphaFold model demonstrates substantially improved accuracy

Josh Abrahamsen^{1,2}, Alexander Pritzel^{1,2}, Michael O’Hearn^{1,2}, David A. K. Hartland^{1,2}, Christopher J. Hays^{1,2}, Zachariah S. Bonsu^{1,2}, David Bartels³, Alex Brigandt⁴, Alenay Chempak⁵, Steven Rivera⁶, Andrew Cowie⁶, Michael Figari⁶, Catherine Young⁶, Kieren K. Khor⁶, Catherine M. R. Lew⁶, Catherine Tong⁶, Sergei Vakser⁶, Ellen S. Zhang⁶, Austin Ziskin⁶, Victor Kapur⁶, Pudvezan Kachli⁶, Max Jaderberg^{6,7}, & John M. Jumper^{1,2}

Probing the effects of broken symmetries in machine learning

Marc F. Langer¹, Sergey N. Prodnikov² and Michele Ceriotti³

Laboratory of Computational Science and Modelling and National Centre for Computational Design and Discovery of Novel Materials MARVEL, Institute of Materials, Ecole Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland

¹ Author to whom any correspondence should be addressed.

E-mail: michele.ceriotti@epfl.ch

Keywords: machine learning, symmetry-constrained models, atomistic modeling, molecular simulations

Supplementary material for this article is available [online](#)

Abstract

Symmetry is one of the most central concepts in physics, and it is no surprise that it has also been widely adopted as an inductive bias for machine-learning models applied to the physical sciences. This is especially true for models targeting the properties of matter at the atomic scale. Both established and state-of-the-art approaches, with almost no exceptions, are built to be exactly equivariant to translations, permutations, and rotations of the atoms. Incorporating symmetries—rotations in particular—constraints the model design space and implies more complicated architectures that are often also computationally demanding. There are indications

The Importance of Being Scalable: Improving the Speed and Accuracy of Neural Network Interatomic Potentials Across Chemical Domains

Eric Qu¹
UC Berkeley
ericqu@berkeley.edu

Aditi S. Krishnapriyan¹
UC Berkeley, LBNL
aditik1@berkeley.edu

Abstract

Scaling has been a critical factor in improving model performance and generalization across various fields of machine learning. It involves how a model’s performance changes with increases in model size or input data, as well as how efficiently computational resources are utilized to support this growth. Despite successes in scaling other types of machine learning models, the study of scaling in Neural Network Interatomic Potentials (NNIPs) remains limited. NNIPs act as

Two for One: Diffusion Models and Force Fields for Coarse-Grained Molecular Dynamics

Marioes Arts,^{1,2,3,4} Victor García Satorras,^{1,5,6} Chin-Wei Huang,¹ Daniel Zügner,⁵ Marco Federici,^{1,3} Cecilia Clementi,^{5,2} Frank Noé,¹ Robert Pinsler,⁶ and Rianne van den Berg¹

¹ Work done during an internship at Microsoft Research (Amsterdam).

² University of Copenhagen, Department of Computer Science, Universitetsparken 1, Copenhagen, 2100, Denmark.

³ AI4Science, Microsoft Research, Evert van de Beekstraat 354, Amsterdam, 1118 CZ, The Netherlands.

⁴ AI4Science, Microsoft Research, Karl-Liebknecht-Straße 32, Berlin, 10178, Germany.

⁵ University of Amsterdam, Information Institute, Science Park 904, Amsterdam, 1098 XH, The Netherlands.

⁶ Freie Universität Berlin, Department of Physics, Arnimallee 12, Berlin, 14195, Germany.

^{1,2} AI4Science, Microsoft Research, 21 Station Road, Cambridge, CB1 2FB, United Kingdom.

³ Equal contribution.

E-mail: ma@di.ku.dk; victorgar@microsoft.com

Abstract

Course-grained (CG) molecular dynamics enables the study of biological processes at temporal and spatial scales that would be intractable at an atomistic resolution. However, accurately learning a CG force field remains a challenge. In this work, we leverage connections between score-based generative models, force fields and molecular

Swallowing the Bitter Pill: Simplified Scalable Conformer Generation

Yuyang Wang¹, Ahmed A. Elbag^{1,2}, Navdeep Jaitly³, Joshua M. Susskind¹, Miguel Ángel Bautista¹

Abstract

We present a novel way to predict molecular conformers through a simple formulation that sidesteps many of the heuristics of prior works and achieves state of the art results by using the advantages of scale. By training a diffusion generative model directly on 3D atomic positions without any constraints about the explicit structure of molecules (e.g., minimum bond angles) we are able to radically simplify structure generation and predict conformers even when

is the vast complexity of the 3D structure space, encompassing factors such as bond lengths and torsional angles. Designing the molecular conformer space with no geometric constraints, such as bond types and spatial arrangements determined by chiral centers, the conformational space experiences exponential growth with the expansion of the graph size and the number of rotatable bonds (Aszkenasy & Gomez-Bombarelli, 2022). This complicates brute force and exhaustive approaches, making them virtually unusable for even moderately small molecules. Systematic methods, like DMFGA (Hawkins et al., 2018),

Data augmentation

Article

Accurate structure prediction of biomolecular interactions with AlphaFold 3

<https://doi.org/10.1038/s43998-024-07487-w>

Received: 19 December 2023

Accepted: 29 April 2024

Published online: 8 May 2024

Open access

Check for updates

No Equivariance

The introduction of AlphaFold 2 has spurred a revolution in modelling the structure of proteins and their interactions, enabling a huge range of applications in protein modelling and design^{1–3}. Here we describe the AlphaFold 3 model with substantially updated architecture and improved accuracy of predicting the three-dimensional structure of complex systems including proteins, nucleic acids, small molecules, ions and modified residues. The new AlphaFold model demonstrates substantially improved accuracy

Josh Abrahamsen^{1,2}, Alexander Pritzel^{1,2}, Michael O’Hearn^{1,2}, David A. K. Hartland^{1,2}, Christopher Chiaradia^{1,2}, Zachary D. Lipton^{1,2}, Pratiksha Bhatnagar¹, Zohreh Soltani-Seyydlou¹, David Bartok^{1,2}, Alex Bridgland¹, Alenay Chempuru¹, Steven Rivera¹, Andrew Cowie¹, Michael Fipplauer¹, Daniel Gómez-Villegas¹, Yannick Kloss¹, Catherine M. R. Lew¹, Daniel P. Pappalardo¹, Subhajit Bhattacharya¹, Aditi S. Patel¹, Catherine Tong¹, Sergei Vakser¹, Ellen S. Zhang¹, Austin Ziskin¹, Victor Kapur¹, Pudhvezar Kathi¹, Max Jaderberg^{1,2}, & John M. Jumper^{1,2}

Probing the effects of broken symmetries in machine learning

Marc F. Langer¹, Sergey N. Prodnikov² and Michele Ceriotti³

Laboratory of Computational Science and Modelling and National Centre for Computational Design and Discovery of Novel Materials MARVEL, Institute of Materials, Ecole Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland

¹ Author to whom any correspondence should be addressed.

E-mail: michele.ceriotti@epfl.ch

Keywords: machine learning, symmetry-constrained models, atomistic modeling, molecular simulations

Supplementary material for this article is available [online](#)

Abstract

Symmetry is one of the most central concepts in physics, and it is no surprise that it has also been widely adopted as an inductive bias for machine-learning models applied to the physical sciences. This is especially true for models targeting the properties of matter at the atomic scale. Both established and state-of-the-art approaches, with almost no exceptions, are built to exactly equivariant to translations, permutations, and rotations of the atoms. Incorporating symmetries—rotations in particular—constraints the model design space and implies more complicated architectures that are often also computationally demanding. There are indications

The Importance of Being Scalable: Improving the Speed and Accuracy of Neural Network Interatomic Potentials Across Chemical Domains

Eric Qu¹
UC Berkeley
ericqu@berkeley.edu

Aditi S. Krishnapriyan¹
UC Berkeley, LBNL
aditik1@berkeley.edu

Abstract

Scaling has been a critical factor in improving model performance and generalization across various fields of machine learning. It involves how a model's performance changes with increases in model size or input data, as well as how efficiently computational resources are utilized to support this growth. Despite successes in scaling other types of machine learning models, the study of scaling in Neural Network Interatomic Potentials (NNIPs) remains limited. NNIPs act as

Two for One: Diffusion Models and Force Fields for Coarse-Grained Molecular Dynamics

Marioes Arts,^{1,1,2,3} Victor García Satorras,^{1,4,5} Chin-Wei Huang,¹ Daniel Zugner,¹ Marco Federici,^{1,1} Cecilia Clementi,^{1,2} Frank Noé,¹ Robert Pinsler,¹ and Rianne van den Berg¹

¹ Work done during an internship at Microsoft Research (Amsterdam).
² University of Copenhagen, Department of Computer Science, Universitetsparken 1, Copenhagen, 2100, Denmark.

³ AI4Science, Microsoft Research, Evert van de Beekstraat 354, Amsterdam, 1118 CZ, The Netherlands.

⁴ AI4Science, Microsoft Research, Karl-Liebknecht-Straße 32, Berlin, 10178, Germany.
⁵ University of Amsterdam, Information Institute, Science Park 904, Amsterdam, 1098 XH, The Netherlands.

¹ Freie Universität Berlin, Department of Physics, Arnimallee 12, Berlin, 14195, Germany.
² AI4Science, Microsoft Research, 21 Station Road, Cambridge, CB1 2FB, United Kingdom.

³ Equal contribution.

* E-mail: ma@di.ku.dk; victorgar@microsoft.com

Abstract

Course-grained (CG) molecular dynamics enables the study of biological processes at temporal and spatial scales that would be intractable at an atomistic resolution. However, accurately learning a CG force field remains a challenge. In this work, we leverage connections between score-based generative models, force fields and molecular

Swallowing the Bitter Pill: Simplified Scalable Conformer Generation

Yuyang Wang¹, Ahmed A. Elbag^{1,2}, Navdeep Jaitly¹, Joshua M. Susskind¹, Miguel Ángel Bautista¹

Abstract

We present a novel way to predict molecular conformers through a simple formulation that sidesteps many of the heuristics of prior works and achieves state of the art results by using the advantages of scale. By training a diffusion generative model directly on 3D atomic positions without any constraints about the chemical structure of molecules (or, in other words, no bond angles) we are able to radically simplify structure generation and make it tractable on much larger molecules.

Systematic methods, like DMFGA (Hawkins et al., 2018),

is the vast complexity of the 3D structure space, encompassing factors such as bond lengths and torsional angles. Designing the molecular conformer space with constraints such as bond types and spatial arrangements determined by chiral centers, the conformational space experiences exponential growth with the expansion of the graph size and the number of rotatable bonds (Aszkenasy & Gomez-Bombarelli, 2022). This complicates brute force and exhaustive approaches, making them virtually unfeasible for even moderately small molecules.

DOES EQUIVARIANCE MATTER AT SCALE?

Johann Bremer¹, Sönke Behrends¹, Pim de Haan², Taco Cohen¹
Quacquarelli AI Research¹
mail@johannbremer.de

ABSTRACT

Given large data sets and sufficient compute, is it beneficial to design neural architectures for the structure and symmetries of each problem? Or is it more efficient to learn them from data? We study empirically how equivariant and non-equivariant networks scale with compute and training samples. Focusing on a benchmark problem of rigid-body interactions and on general-purpose transformer architectures, we perform a series of experiments, varying the model size, training steps, and dataset size. We find that non-equivariant models with data augmentation are less efficient, but training non-equivariant models with data augmentation can close this gap given sufficient epochs. Second, scaling with compute follows a power law, with equivariant models outperforming non-equivariant ones at each tested compute budget. Finally, the optimal allocation of a compute budget onto model size and training duration differs between equivariant and non-equivariant models.

Data augmentation

- thumb-up Easy to implement
- thumb-up 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?

Empirical NTK

Training dynamics under continuous gradient descent:

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

learning rate

loss

training sample

The diagram illustrates the components of the training dynamics equation. It shows a blue arrow pointing from the text 'learning rate' to the scalar factor $-\frac{\eta}{N}$. Another blue arrow points from the text 'loss' to the term $\frac{\partial L}{\partial \mathcal{N}(x_i)}$. A third blue arrow points from the text 'training sample' to the summation index $i=1$ in the term $\sum_{i=1}^N$.

Empirical NTK

Training dynamics under continuous gradient descent:

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

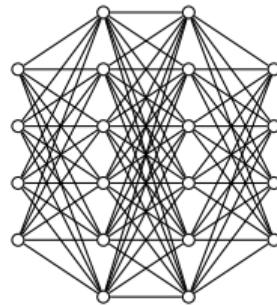
↑
learning rate ↑
↑
training sample ↑
loss

with the **empirical neural tangent kernel (NTK)**

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

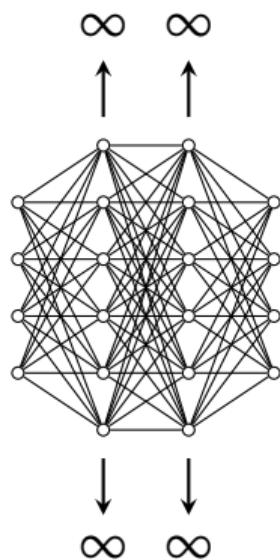
Infinite width limit

[Jacot et al. 2018]



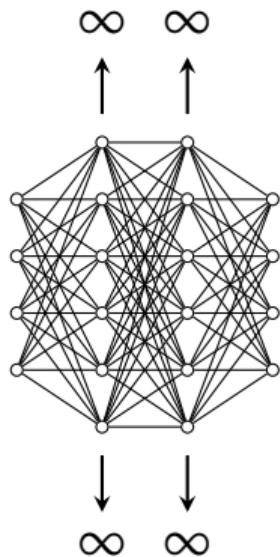
Infinite width limit

[Jacot et al. 2018]



Infinite width limit

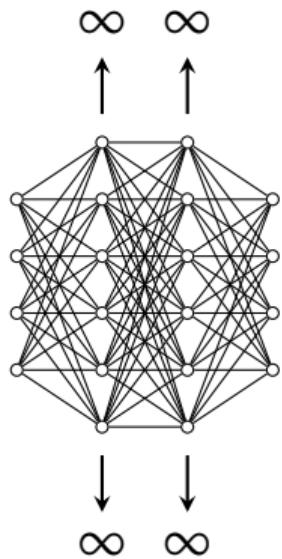
[Jacot et al. 2018]



👍 NTK becomes independent of initialization

Infinite width limit

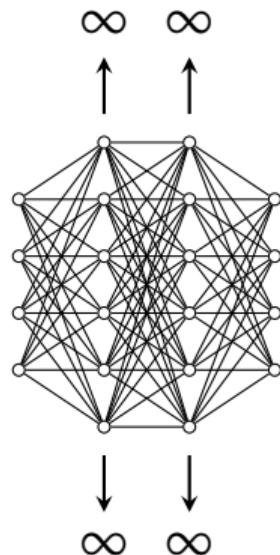
[Jacot et al. 2018]



- 👍 NTK becomes independent of initialization
- 👍 NTK becomes constant in training

Infinite width limit

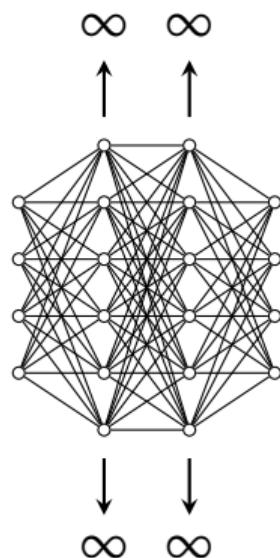
[Jacot et al. 2018]



- 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]

- ① 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]

- ① At infinite width, the mean prediction is given by

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

neural tangent kernel



Mean prediction from NTK

[Jacot et al. 2018]

- ① At infinite width, the mean prediction is given by

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

neural tangent kernel

train data

Mean prediction from NTK

[Jacot et al. 2018]

- ① At infinite width, the mean prediction is given by

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

neural tangent kernel

learning rate

train data

```
graph TD; NTK[neural tangent kernel] --> XxTheta[Θ(x, X)]; LR[learning rate] --> expTerm[e^{-ηΘ(X, X)t}]; TrainData[train data] --> Y[Y];
```

Mean prediction from NTK

[Jacot et al. 2018]

- ① At infinite width, the mean prediction is given by

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

Diagram illustrating the components of the mean prediction formula:

- neural tangent kernel**: Points to the term $\Theta(x, X)$.
- train labels**: Points to the term Y .
- learning rate**: Points to the term $e^{-\eta\Theta(X, X)t}$.
- train data**: Points to the term $\Theta(X, X)^{-1}$.

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

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

The diagram shows the mathematical expression for augmented data $\mu_t(x)$. It consists of several blue arrows pointing from labels to terms in the equation:

- An arrow points from the label "augmented data" to the term $\Theta(x, X)$.
- Two arrows point from the label "augmented data" to the term $\Theta(X, X)^{-1}$.
- Two arrows point from the label "augmented labels" to the term $(\mathbb{I} - e^{-\eta\Theta(X, X)t})Y$.
- A single arrow points from the label "augmented labels" to the label "augmented data".

Data augmentation at infinite width

group transformation

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

augmented data augmented labels

The diagram illustrates the group transformation equation. A blue arrow labeled "group transformation" points to the equation. Another blue arrow labeled "augmented data" points to the term $\Theta(\rho(g)x, X)$. A third blue arrow labeled "augmented labels" points to the term Y . The equation itself is
$$\mu_t(\rho(g)x) = \Theta(\rho(g)x, X)\Theta(X, X)^{-1}(\mathbb{I} - e^{-\eta}\Theta(X, X)t)Y$$
.

Kernel transformation

The neural tangent kernel Θ as well as the NNGP kernel K transform according to

$$\begin{aligned}\Theta(\rho(g)x, \rho(g)x') &= \rho_K(g)\Theta(x, x')\rho_K^\top(g), \\ K(\rho(g)x, \rho(g)x') &= \rho_K(g)K(x, x')\rho_K^\top(g),\end{aligned}$$

for all $g \in G$ and $x, x' \in X$.

Kernel transformation

The neural tangent kernel Θ as well as the NNGP kernel K transform according to

$$\begin{aligned}\Theta(\rho(g)x, \rho(g)x') &= \rho_K(g)\Theta(x, x')\rho_K^\top(g), \\ K(\rho(g)x, \rho(g)x') &= \rho_K(g)K(x, x')\rho_K^\top(g),\end{aligned}$$

for all $g \in G$ and $x, x' \in X$.

Hence, for MLPs,

$$\Theta(\rho(g)x, \rho(g)x') = \Theta(x, x') \quad \Rightarrow \quad \Theta(\rho(g)x, x') = \Theta(x, \rho^{-1}(g)x')$$

Permutation shift

- On the training data, group transformations permute the samples

$$\rho(g)x_i = x_{\pi_g(i)}, \quad \pi_g \in S_N$$

Permutation shift

- On the training data, group transformations permute the samples

$$\rho(g)x_i = x_{\pi_g(i)}, \quad \pi_g \in S_N$$

- Therefore, for a permutation of training samples associate to g

$$\begin{aligned}\Pi(g)\Theta(X, X) &= \Theta(\rho(g)X, X) \\ &= \Theta(X, \rho^{-1}(g)X) \\ &= \Theta(X, X)(\Pi^{-1}(g))^\top \\ &= \Theta(X, X)\Pi(g)\end{aligned}$$

Data augmentation at infinite width

group transformation

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

augmented data augmented labels

The diagram illustrates the mathematical expression for data augmentation. A blue curved arrow labeled "group transformation" points to the term $\rho(g)x$. Another blue curved arrow labeled "augmented data" points to the leftmost term $\rho(g)x$. A third blue curved arrow labeled "augmented labels" points to the rightmost term Y .

Data augmentation at infinite width

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

group transformation

for augmented data

augmented data

augmented labels

The diagram illustrates the components of the data augmentation formula. A large blue oval encloses the right-hand side of the equation. Inside the oval, the term $\Theta(X, X)^{-1}$ is highlighted with a red bracket. Four blue arrows point from labels below the oval to different parts of the equation: one to $\rho(g)$, one to X , one to $\Theta(X, X)^{-1}$, and one to Y . Another blue arrow points from the label "group transformation" to the left side of the equation. The label "augmented data" is placed under the term $\Theta(\rho(g)x, X)$, and the label "augmented labels" is placed under the term $\Theta(X, X)t)Y$.

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

The diagram illustrates the components of the group transformation equation. A blue arrow labeled "group transformation" points to the right side of the equation. Another blue arrow labeled "augmented data" points to the term $\Theta(x, X)$. A third blue arrow labeled "augmented labels" points to the term Y .

Data augmentation at infinite width

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

group transformation

augmented labels

Data augmentation at infinite width

group transformation

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

for invariance

Mean prediction

$$\mu_t(x)$$

Mean prediction

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

Mean prediction

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

Mean prediction

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

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

Main conclusion

Deep ensembles trained with data augmentation are equivariant.

- ✓ Proof of exact equivariance for
 - full data augmentation
 - infinite ensembles
- ✓ 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
- ✓ Equivariance holds for all training times
- ✓ Equivariance holds away from the training data

Main conclusion

Deep ensembles trained with data augmentation are equivariant.

- ✓ Proof of exact equivariance for
 - full data augmentation
 - infinite ensembles
- ✓ Equivariance holds for all training times
- ✓ Equivariance holds away from the training data
- ✓ Holds also for finite-width networks

[Nordenfors, Flinth 2024]

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

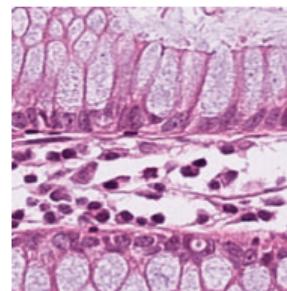
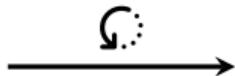
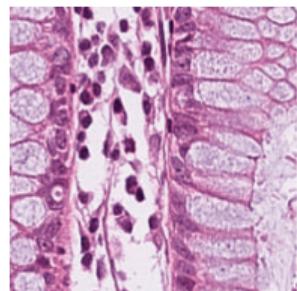
Intuitive explanation

- ✓ Equivariance holds for all training times
 - ✓ Equivariance holds away from the training data
-
- ➊ At infinite width, the mean output at initialization is zero everywhere.
 - ⇒ Training with full data augmentation leads to an equivariant function.

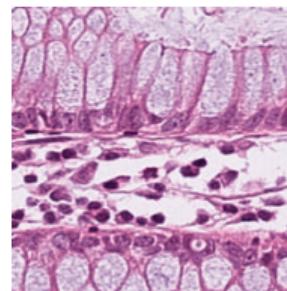
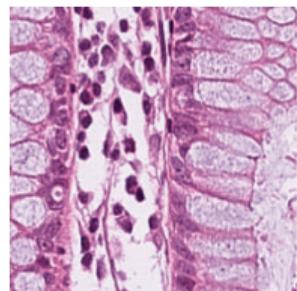
What Does An Augmented Ensemble Converge To?

Rotating images

Rotating images



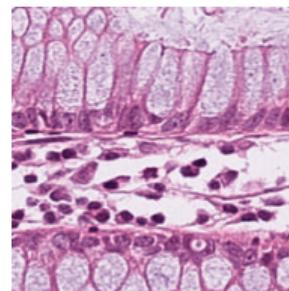
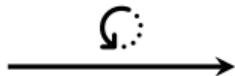
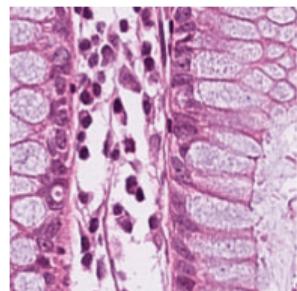
Rotating images



$$f(x)$$

$f : \text{pixels} \rightarrow \text{colors}$

Rotating images



$$\begin{array}{c} f(x) \\ f : \text{pixels} \rightarrow \text{colors} \end{array}$$



$$\begin{aligned} f(\rho(g^{-1})x) \\ = [\rho_{\text{reg}}(g)f](x) \end{aligned}$$

Data augmentation and NTKs

Data augmentation and NTKs

Consider two ensembles:

trained without data augmentation

trained with data augmentation

Data augmentation and NTKs

Consider two ensembles:

trained without data augmentation

trained with data augmentation

If

$$\Theta^{\text{non-aug}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{aug}}(f, \rho_{\text{reg}}(g)f')$$

Data augmentation and NTKs

Consider two ensembles:

trained without data augmentation

trained with data augmentation

If

$$\Theta^{\text{non-aug}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{aug}}(f, \rho_{\text{reg}}(g)f')$$

Then

$$\mu_t^{\text{non-aug}}(x) = \mu_t^{\text{aug}}(x)$$

at infinite width.

Data augmentation and NTKs

Consider two ensembles:

trained without data augmentation

trained with data augmentation

If

$$\Theta^{\text{non-aug}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{aug}}(f, \rho_{\text{reg}}(g)f')$$

Then

$$\mu_t^{\text{non-aug}}(x) = \mu_t^{\text{aug}}(x) \quad \forall t$$

at infinite width.

Data augmentation and NTKs

Consider two ensembles:

trained without data augmentation

trained with data augmentation

If

$$\Theta^{\text{non-aug}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{aug}}(f, \rho_{\text{reg}}(g)f')$$

Then

$$\mu_t^{\text{non-aug}}(x) = \mu_t^{\text{aug}}(x) \quad \forall t \quad \forall x$$

at infinite width.

Data augmentation and NTKs

$$\Theta^{\text{non-aug}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{aug}}(f, \rho_{\text{reg}}(g)f')$$

Data augmentation and NTKs

$$\Theta^{\text{non-aug}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{aug}}(f, \rho_{\text{reg}}(g)f')$$

- ① Given an architecture with NTK Θ^{aug} ,
find an architecture with NTK $\Theta^{\text{non-aug}}$

Group convolutions

[Cohen, Welling 2016]

Group convolutions

[Cohen, Welling 2016]

Group conv's are the (unique) linear layers equivariant wrt ρ_{reg}

Group convolutions

[Cohen, Welling 2016]

Group conv's are the (unique) linear layers equivariant wrt ρ_{reg}

- Ordinary convolutions

$$f'(y) = \int_X dx \kappa(x - y) f(x)$$

Group convolutions

[Cohen, Welling 2016]

Group conv's are the (unique) linear layers equivariant wrt ρ_{reg}

- Ordinary convolutions

$$f'(y) = \int_X dx \kappa(x - y) f(x)$$

- Group convolutions

$$f'(g) = \int_X dx \kappa(\rho(g^{-1})x) f(x) \quad \text{lifting}$$

Group convolutions

[Cohen, Welling 2016]

Group conv's are the (unique) linear layers equivariant wrt ρ_{reg}

- Ordinary convolutions

$$f'(y) = \int_X dx \kappa(x - y) f(x)$$

- Group convolutions

$$f'(g) = \int_X dx \kappa(\rho(g^{-1})x) f(x) \quad \text{lifting}$$

$$f'(g) = \int_G dg \kappa(g^{-1}h) f(h) \quad \text{group convolution}$$

Group convolutions

[Cohen, Welling 2016]

Group conv's are the (unique) linear layers equivariant wrt ρ_{reg}

- Ordinary convolutions

$$f'(y) = \int_X dx \kappa(x - y) f(x)$$

- Group convolutions

$$f'(g) = \int_X dx \kappa(\rho(g^{-1})x) f(x) \quad \text{lifting}$$

$$f'(g) = \int_G dg \kappa(g^{-1}h) f(h) \quad \text{group convolution}$$

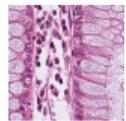
$$f' = \frac{1}{\text{vol}(G)} \int_G dg f(g) \quad \text{group pooling}$$

GCNNs

Stack GConv-layers to obtain an invariant network

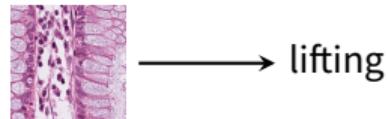
GCNNs

Stack GConv-layers to obtain an invariant network



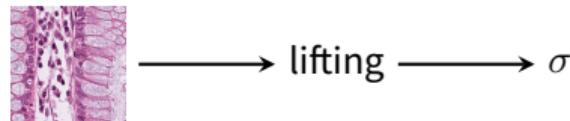
GCNNs

Stack GConv-layers to obtain an invariant network



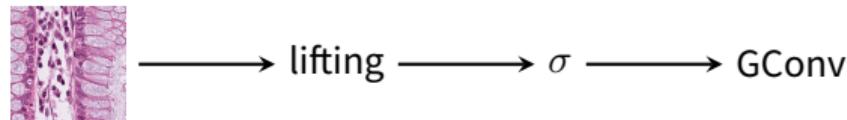
GCNNs

Stack GConv-layers to obtain an invariant network



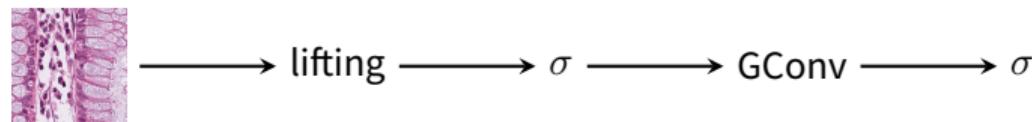
GCNNs

Stack GConv-layers to obtain an invariant network



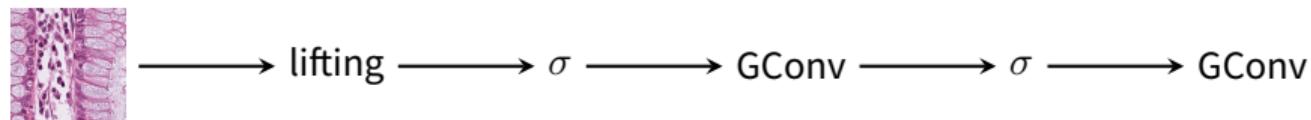
GCNNs

Stack GConv-layers to obtain an invariant network



GCNNs

Stack GConv-layers to obtain an invariant network



GCNNs

Stack GConv-layers to obtain an invariant network



NTKs for GCNNs

For GCNN-layers, define the NNGP and NTK via

$$K_{\mathbf{g}, \mathbf{g}'}^{(\ell)}(\mathbf{f}, \mathbf{f}') = \mathbb{E} \left[[z^{(\ell)}(\mathbf{f})](\mathbf{g}) \left([z^{(\ell)}(\mathbf{f}')] (\mathbf{g}') \right)^\top \right]$$

NTKs for GCNNs

For GCNN-layers, define the NNGP and NTK via

$$K_{\mathbf{g}, \mathbf{g}'}^{(\ell)}(\mathbf{f}, \mathbf{f}') = \mathbb{E} \left[[z^{(\ell)}(\mathbf{f})](\mathbf{g}) \left([z^{(\ell)}(\mathbf{f}')](\mathbf{g}') \right)^\top \right]$$

$$\Theta_{\mathbf{g}, \mathbf{g}'}^{(\ell)}(\mathbf{f}, \mathbf{f}') = \mathbb{E} \left[\sum_{\ell'=1}^{\ell} \frac{\partial [z^{(\ell)}(\mathbf{f})](\mathbf{g})}{\partial \theta^{(\ell')}} \left(\frac{\partial [z^{(\ell)}(\mathbf{f}')](\mathbf{g}')}{\partial \theta^{(\ell')}} \right)^\top \right]$$

NTKs for GCNNs

$$[z^{(\ell)}(f)](g) = \int_G dg \kappa(g^{-1}h) [z^{(\ell-1)}(f)](h)$$

The layer-recursion for a GCNN-layer is given by

$$K_{g,g'}^{(\ell+1)}(f, f') = \frac{1}{|S_\kappa|} \int_{S_\kappa} dh K_{gh,g'h}^{(\ell)}(f, f')$$

NTKs for GCNNs

$$[z^{(\ell)}(f)](g) = \int_G dg \kappa(g^{-1}h) [z^{(\ell-1)}(f)](h)$$

The layer-recursion for a GCNN-layer is given by

$$K_{g,g'}^{(\ell+1)}(f, f') = \frac{1}{|S_K|} \int_{S_K} dh K_{gh,g'h}^{(\ell)}(f, f')$$

$$\Theta_{g,g'}^{(\ell+1)}(f, f') = K_{g,g'}^{(\ell+1)}(f, f') + \frac{1}{|S_K|} \int_{S_K} dh \Theta_{gh,g'h}^{(\ell)}(f, f')$$

GCNNs

Stack GConv-layers to obtain an invariant network



NTKs for GCNNs

Stack GConv-layers to obtain an invariant network



Compute NTK with layer-wise recursion

NTKs for GCNNs

Stack GConv-layers to obtain an invariant network



Compute NTK with layer-wise recursion

0

NTKs for GCNNs

Stack GConv-layers to obtain an invariant network



Compute NTK with layer-wise recursion

$$0 \longrightarrow \Theta_{g,g'}^{(1)}(f,f')$$

NTKs for GCNNs

Stack GConv-layers to obtain an invariant network



Compute NTK with layer-wise recursion

$$0 \longrightarrow \Theta_{g,g'}^{(1)}(f,f') \longrightarrow \Theta_{g,g'}^{(2)}(f,f')$$

NTKs for GCNNs

Stack GConv-layers to obtain an invariant network



Compute NTK with layer-wise recursion

$$0 \longrightarrow \Theta_{g,g'}^{(1)}(f,f') \longrightarrow \Theta_{g,g'}^{(2)}(f,f') \longrightarrow \Theta_{g,g'}^{(3)}(f,f')$$

NTKs for GCNNs

Stack GConv-layers to obtain an invariant network



Compute NTK with layer-wise recursion

$$0 \longrightarrow \Theta_{g,g'}^{(1)}(f,f') \longrightarrow \Theta_{g,g'}^{(2)}(f,f') \longrightarrow \Theta_{g,g'}^{(3)}(f,f') \longrightarrow \Theta_{g,g'}^{(4)}(f,f')$$

NTKs for GCNNs

Stack GConv-layers to obtain an invariant network



Compute NTK with layer-wise recursion

$$0 \longrightarrow \Theta_{g,g'}^{(1)}(f,f') \longrightarrow \Theta_{g,g'}^{(2)}(f,f') \longrightarrow \Theta_{g,g'}^{(3)}(f,f') \longrightarrow \Theta_{g,g'}^{(4)}(f,f') \longrightarrow \Theta_{g,g'}^{(5)}(f,f')$$

NTKs for GCNNs

Stack GConv-layers to obtain an invariant network



Compute NTK with layer-wise recursion

$$0 \longrightarrow \Theta_{g,g'}^{(1)}(f,f') \longrightarrow \Theta_{g,g'}^{(2)}(f,f') \longrightarrow \Theta_{g,g'}^{(3)}(f,f') \longrightarrow \Theta_{g,g'}^{(4)}(f,f') \longrightarrow \Theta_{g,g'}^{(5)}(f,f') \longrightarrow \Theta(f,f')$$

NTKs of MLPs and GCNNs

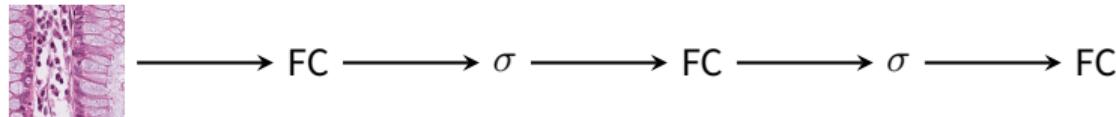
NTKs of MLPs and GCNNs

- Consider two neural networks

NTKs of MLPs and GCNNs

- Consider two neural networks

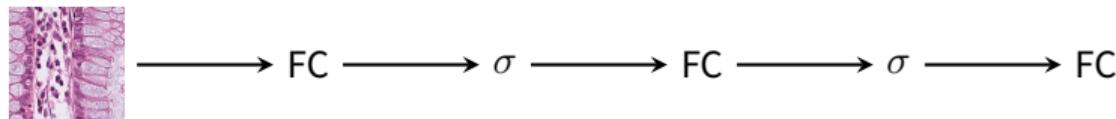
An MLP



NTKs of MLPs and GCNNs

- Consider two neural networks

An MLP



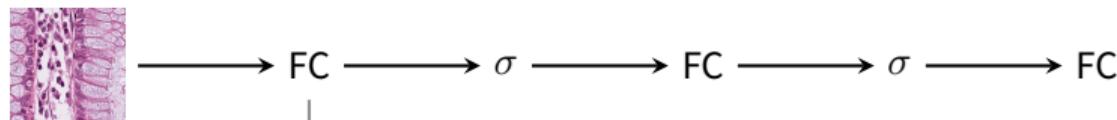
A GCNN



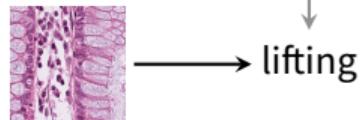
NTKs of MLPs and GCNNs

- Consider two neural networks

An MLP



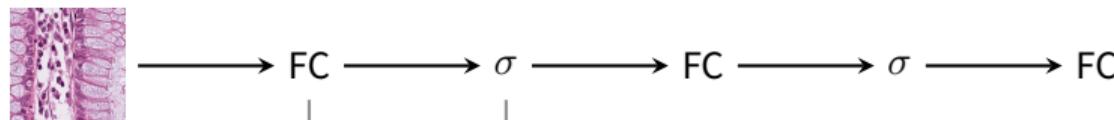
A GCNN



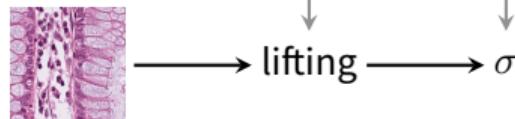
NTKs of MLPs and GCNNs

- Consider two neural networks

An MLP



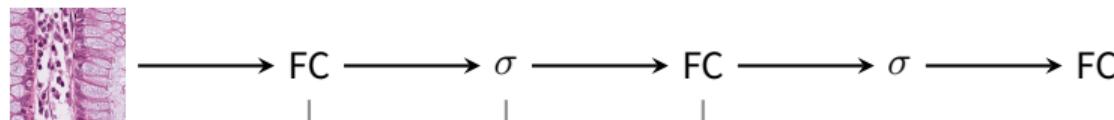
A GCNN



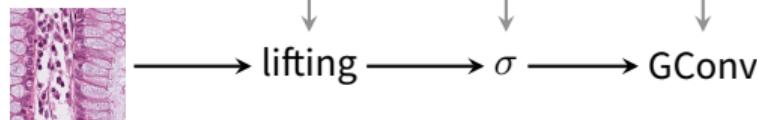
NTKs of MLPs and GCNNs

- Consider two neural networks

An MLP



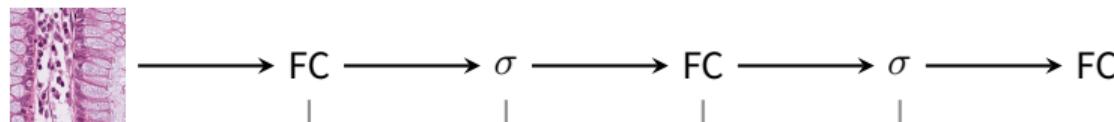
A GCNN



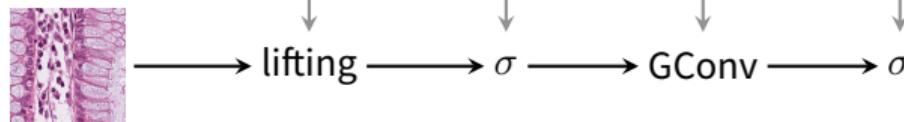
NTKs of MLPs and GCNNs

- Consider two neural networks

An MLP



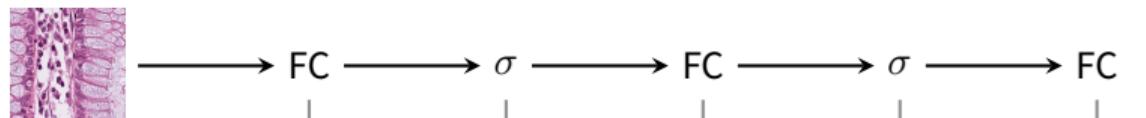
A GCNN



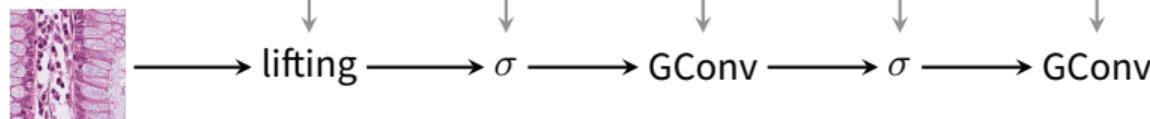
NTKs of MLPs and GCNNs

- Consider two neural networks

An MLP



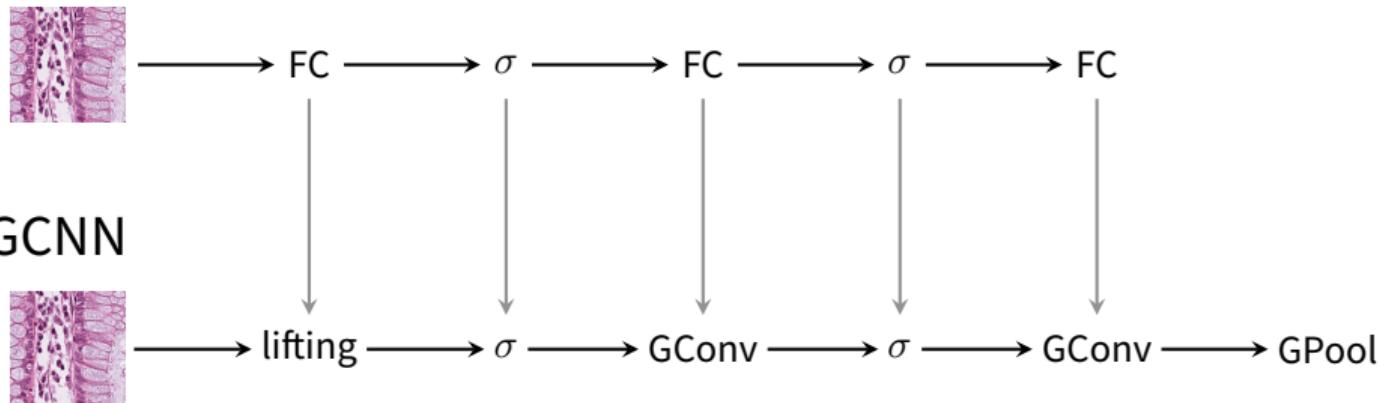
A GCNN



NTKs of MLPs and GCNNs

- Consider two neural networks

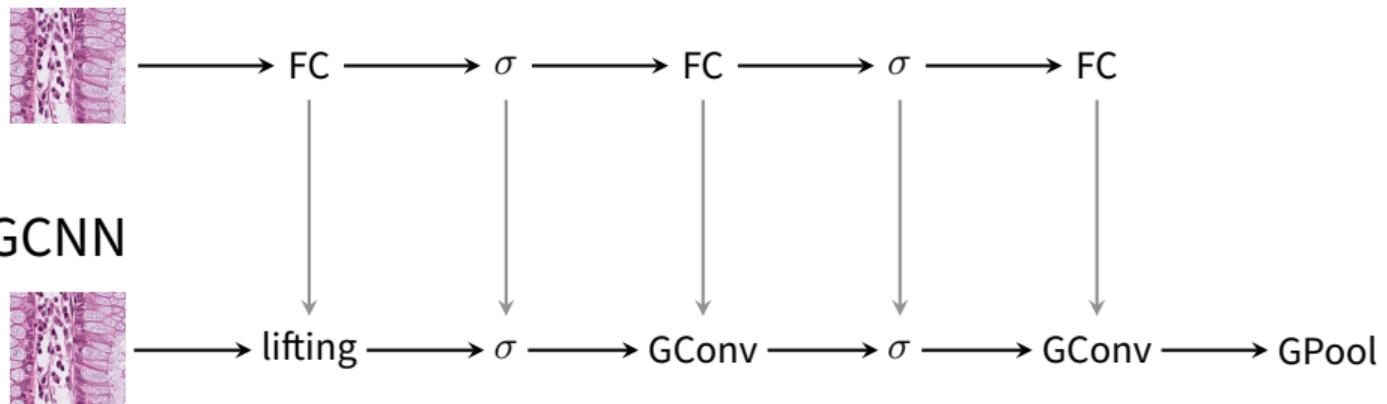
An MLP



NTKs of MLPs and GCNNs

- Consider two neural networks

An MLP



- Then

$$\Theta^{\text{GCNN}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{MLP}}(f, \rho_{\text{reg}}(g)f')$$

Data augmentation of MLPs

$$\Theta^{\text{GCNN}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{MLP}}(f, \rho_{\text{reg}}(g)f')$$

Data augmentation of MLPs

before: non-aug

$$\Theta^{\text{GCNN}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{MLP}}(f, \rho_{\text{reg}}(g)f')$$

Data augmentation of MLPs

before: non-aug

$$\Theta^{\text{GCNN}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{MLP}}(f, \rho_{\text{reg}}(g)f')$$

before: aug

Data augmentation of MLPs

before: non-aug

$$\Theta^{\text{GCNN}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{MLP}}(f, \rho_{\text{reg}}(g)f')$$

before: aug

- ⇒ training the MLP on
G-augmented data

Data augmentation of MLPs

$$\Theta^{\text{GCNN}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{MLP}}(f, \rho_{\text{reg}}(g)f')$$

→ training the MLP on G -augmented data = training the GCNN on unaugmented data

Data augmentation of MLPs

before: non-aug

$$\Theta^{\text{GCNN}}(f, f') = \frac{1}{|G|} \sum_{g \in G} \Theta^{\text{MLP}}(f, \rho_{\text{reg}}(g)f')$$

before: aug

⇒ training the MLP on
G-augmented data

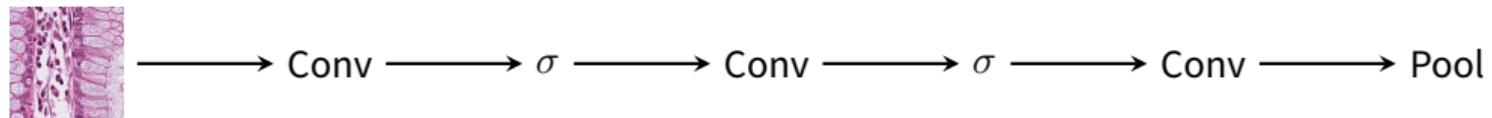
= training the GCNN on
unaugmented data

in the ensemble mean, $\forall t, \forall x$

Data augmentation of CNNs

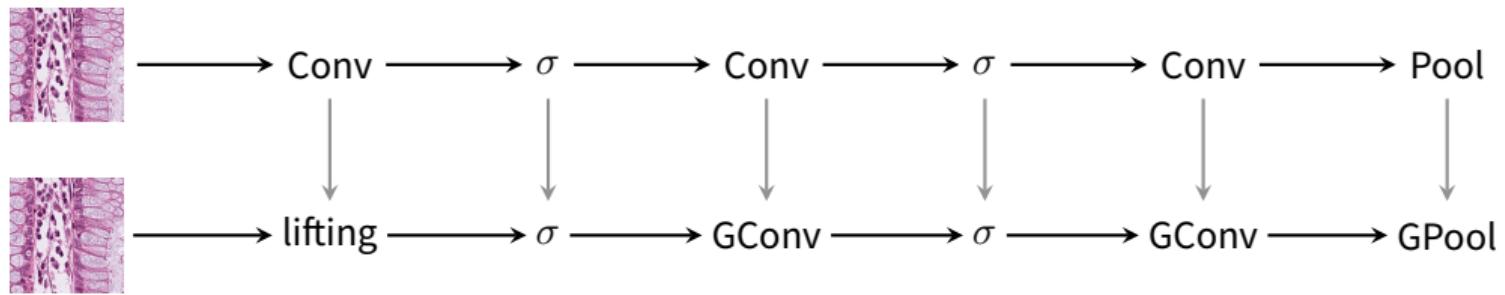
Data augmentation of CNNs

- Consider a CNN



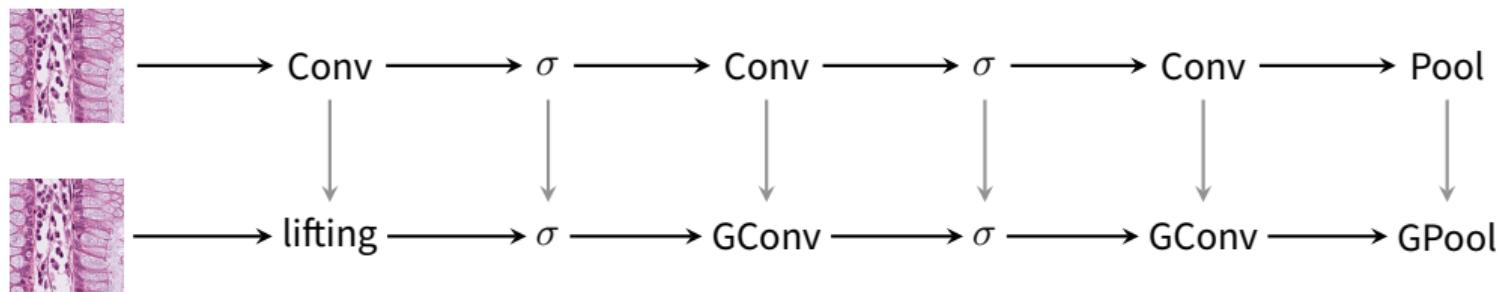
Data augmentation of CNNs

- Consider a CNN and a GCNN invariant wrt. roto-translations



Data augmentation of CNNs

- Consider a CNN and a GCNN invariant wrt. roto-translations

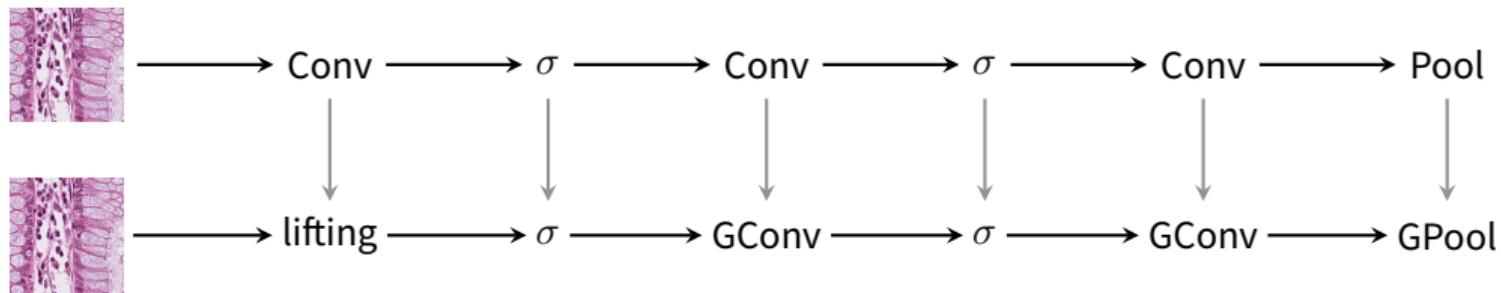


- Then

$$\Theta^{\text{GCNN}}(f, f') = \frac{1}{n} \sum_{r \in C_n} \Theta^{\text{CNN}}(f, \rho_{\text{reg}}(r)f')$$

Data augmentation of CNNs

- Consider a CNN and a GCNN invariant wrt. roto-translations



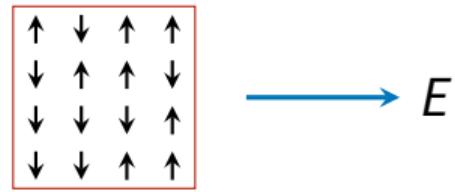
- Then

$$\Theta^{\text{GCNN}}(f, f') = \frac{1}{n} \sum_{r \in C_n} \Theta^{\text{CNN}}(f, \rho_{\text{reg}}(r)f')$$

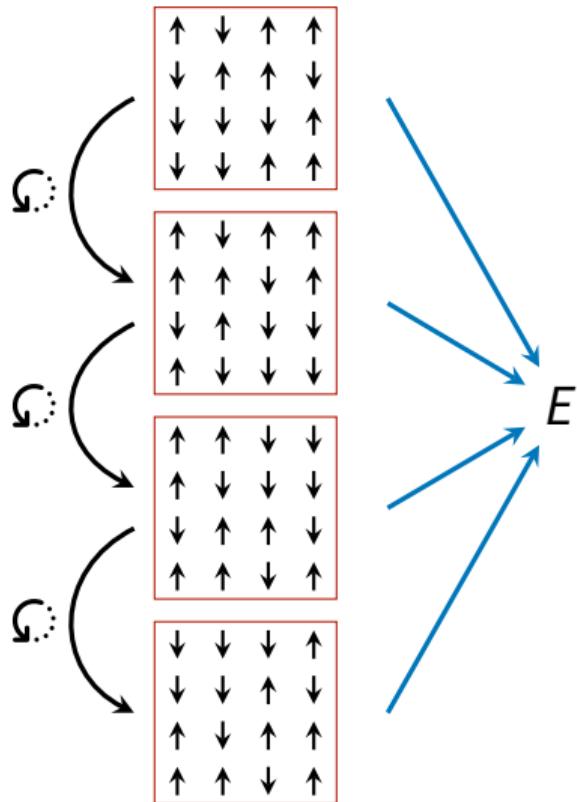
- By training the CNN on rotated images, one obtains a roto-translation invariant GCNN

Experiments

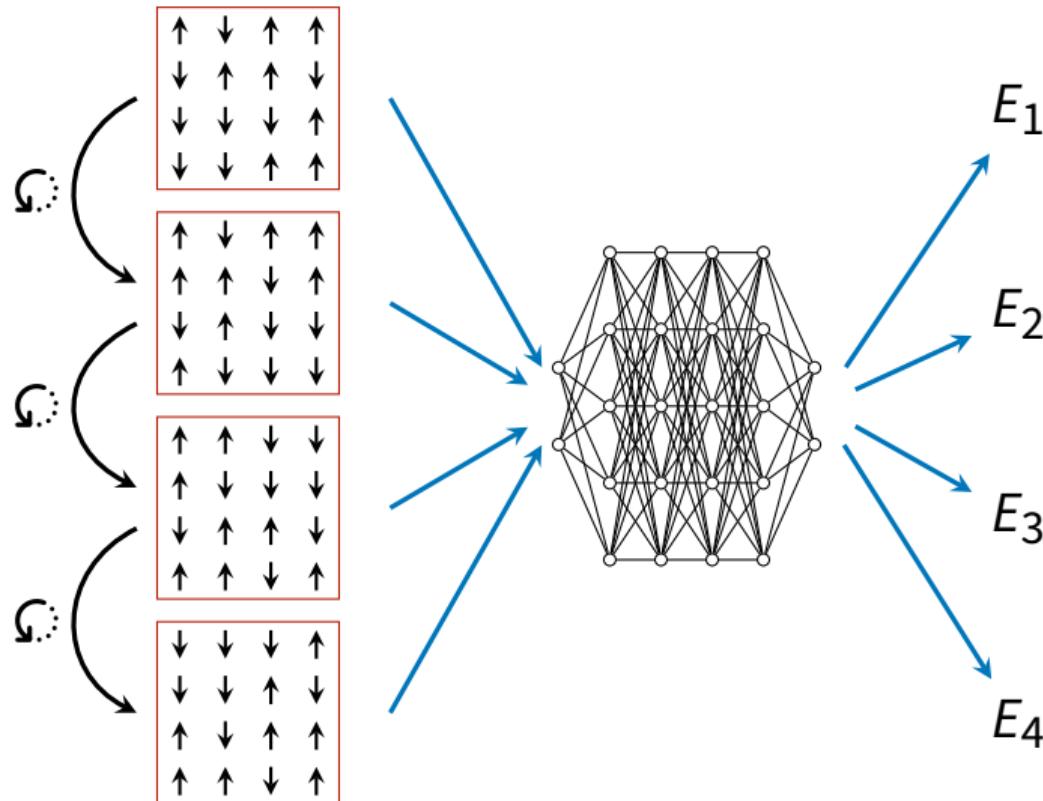
Ising model



Ising model

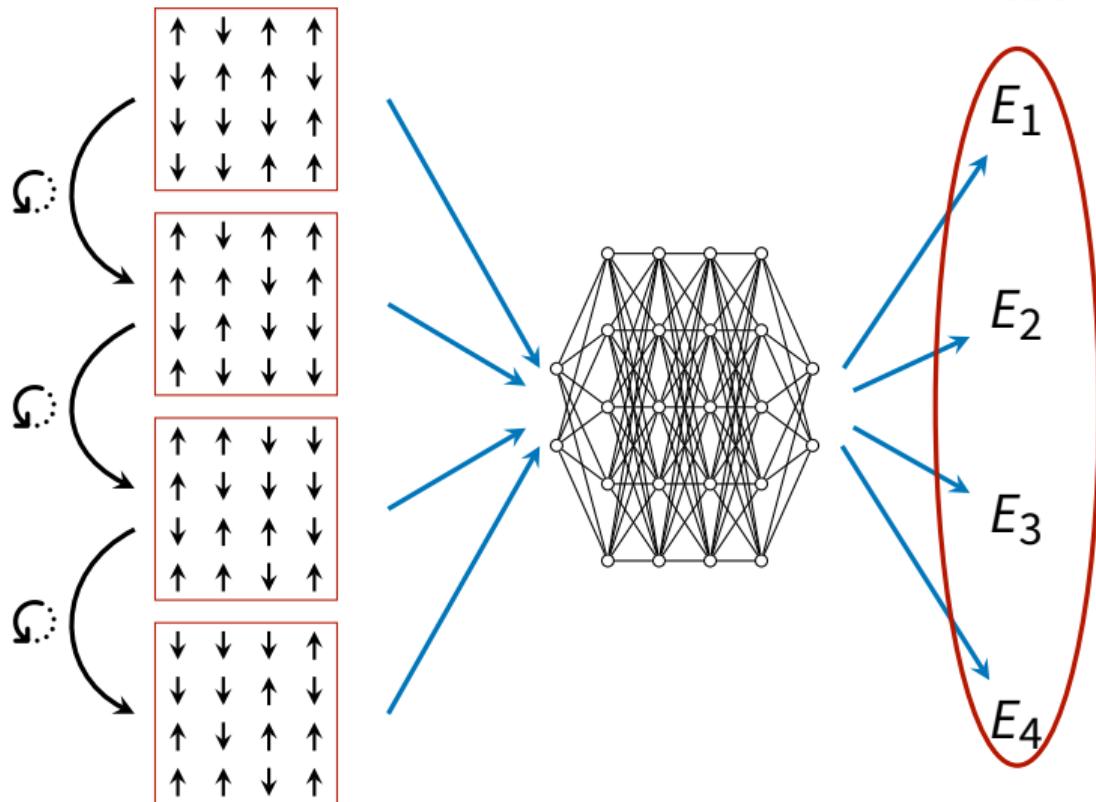


Ising model

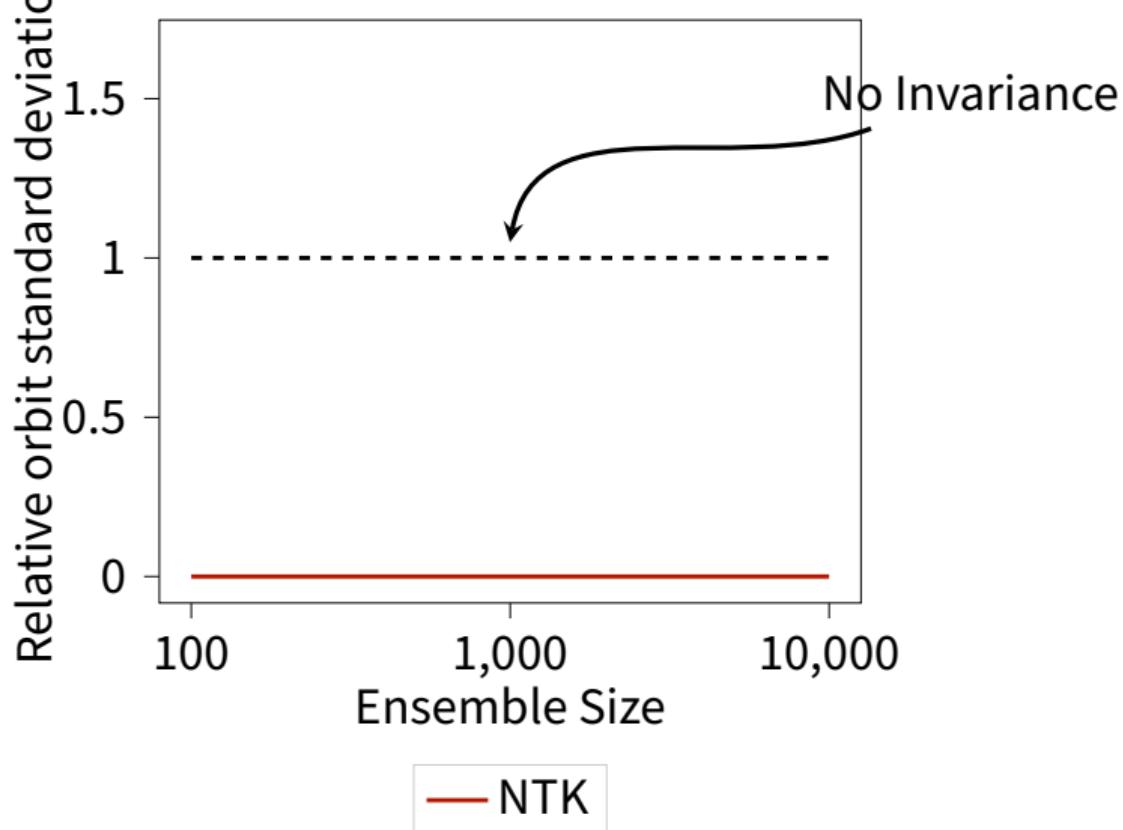


Ising model

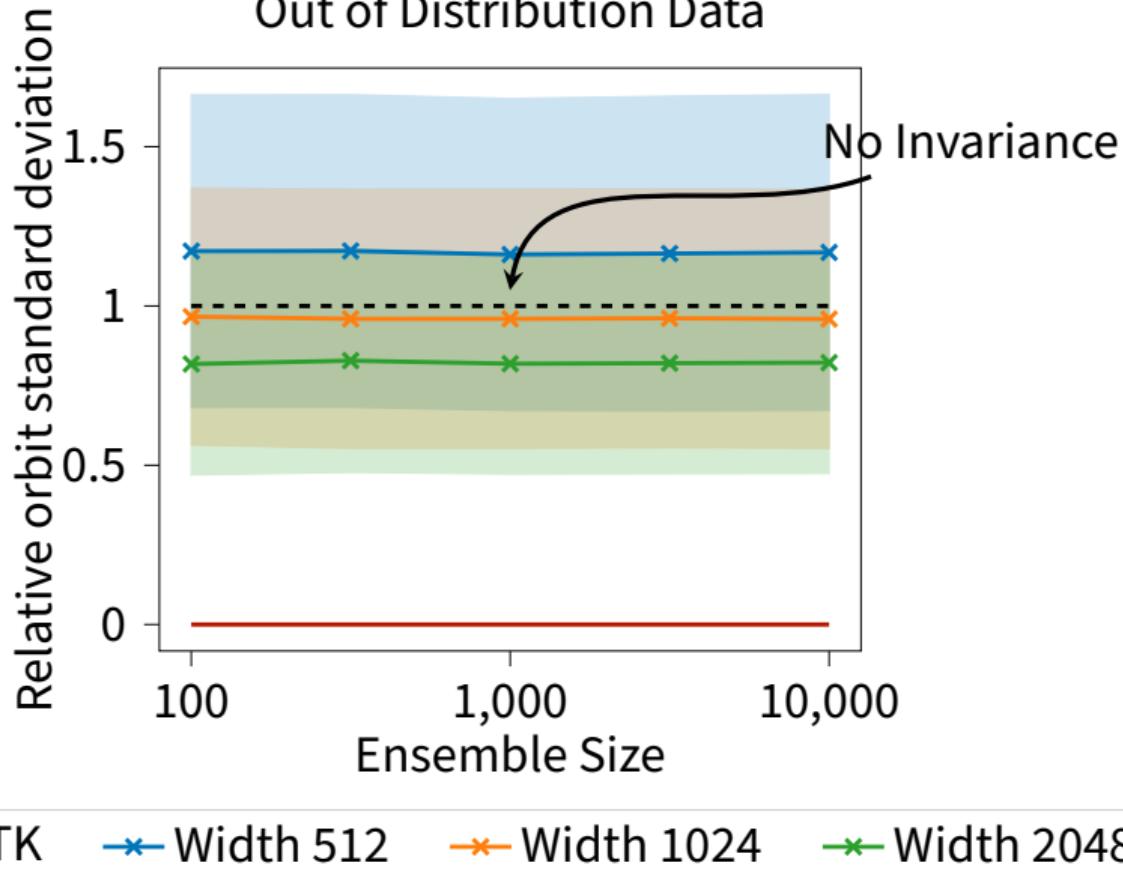
Relative Standard Deviation

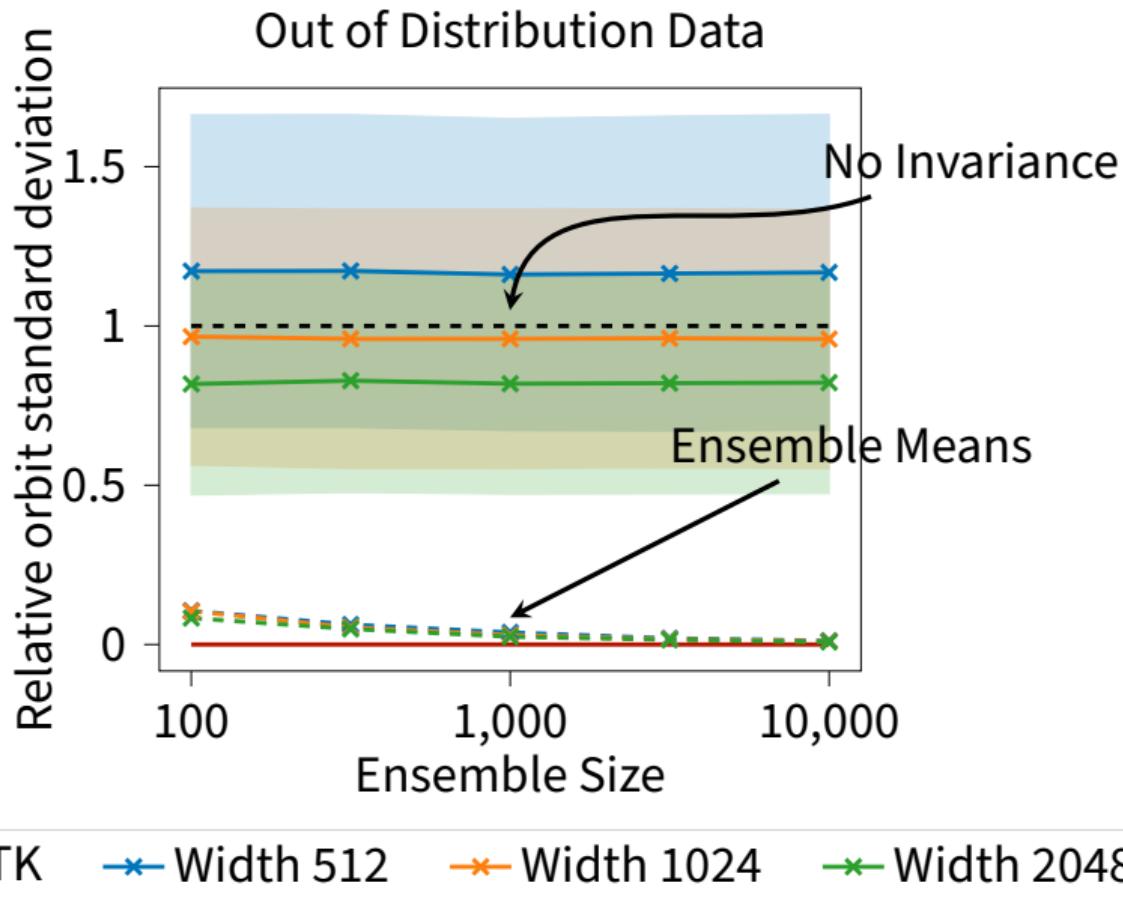


Out of Distribution Data

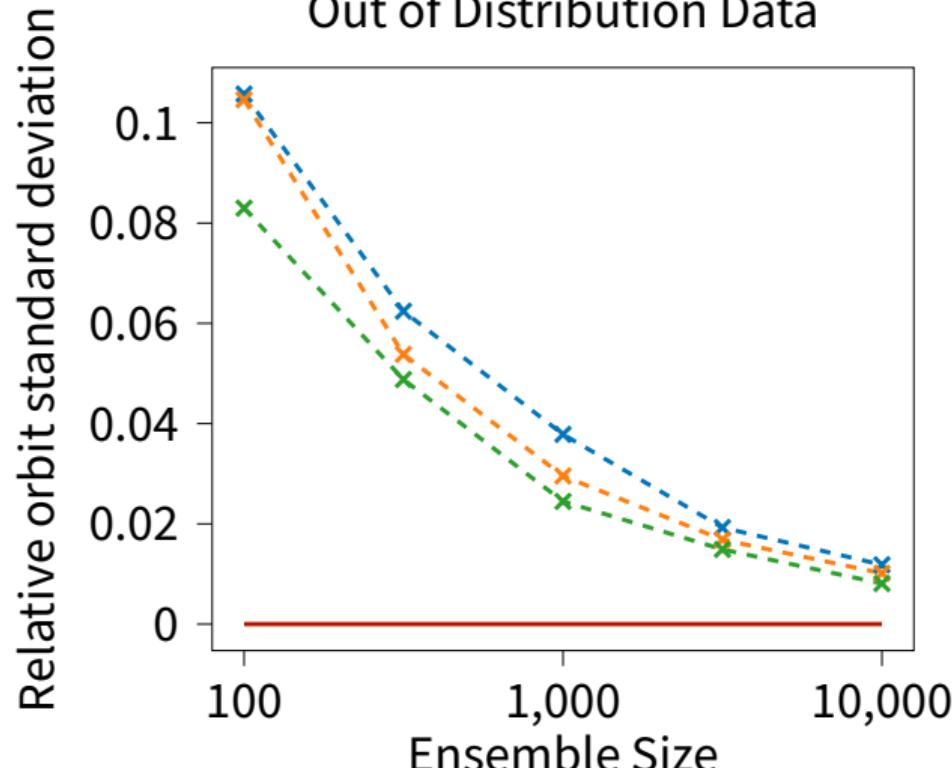


Out of Distribution Data





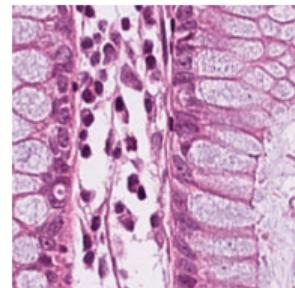
Out of Distribution Data



— NTK -x- Width 512 -x- Width 1024 -x- Width 2048

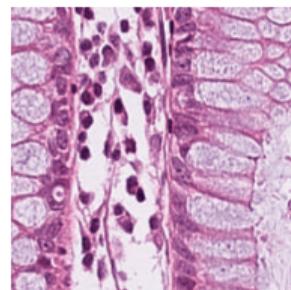
Histological slices

[Kather et al. 2018]



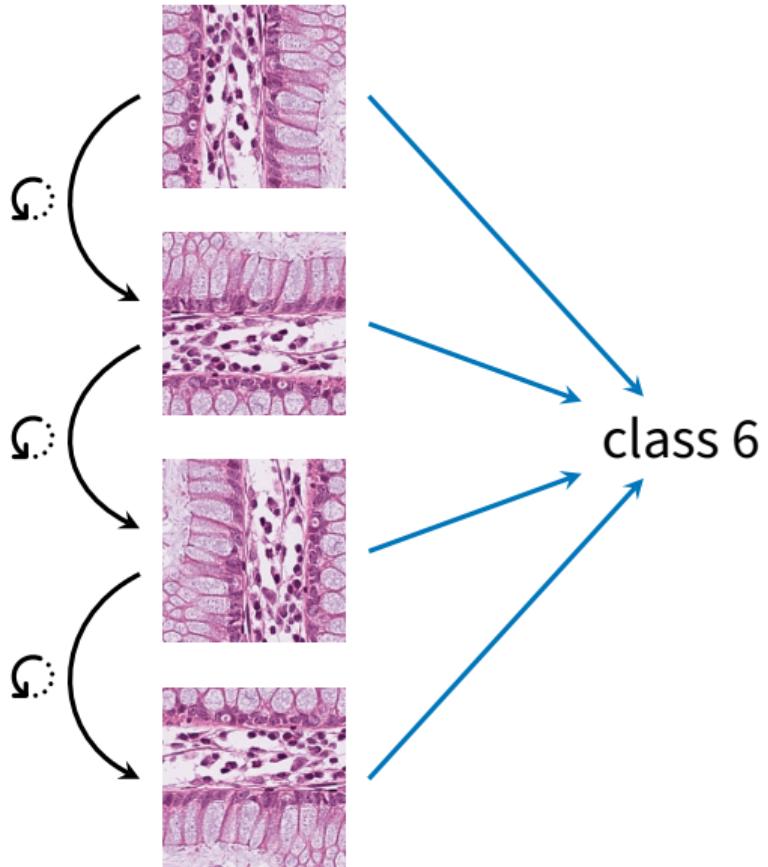
Histological slices

[Kather et al. 2018]

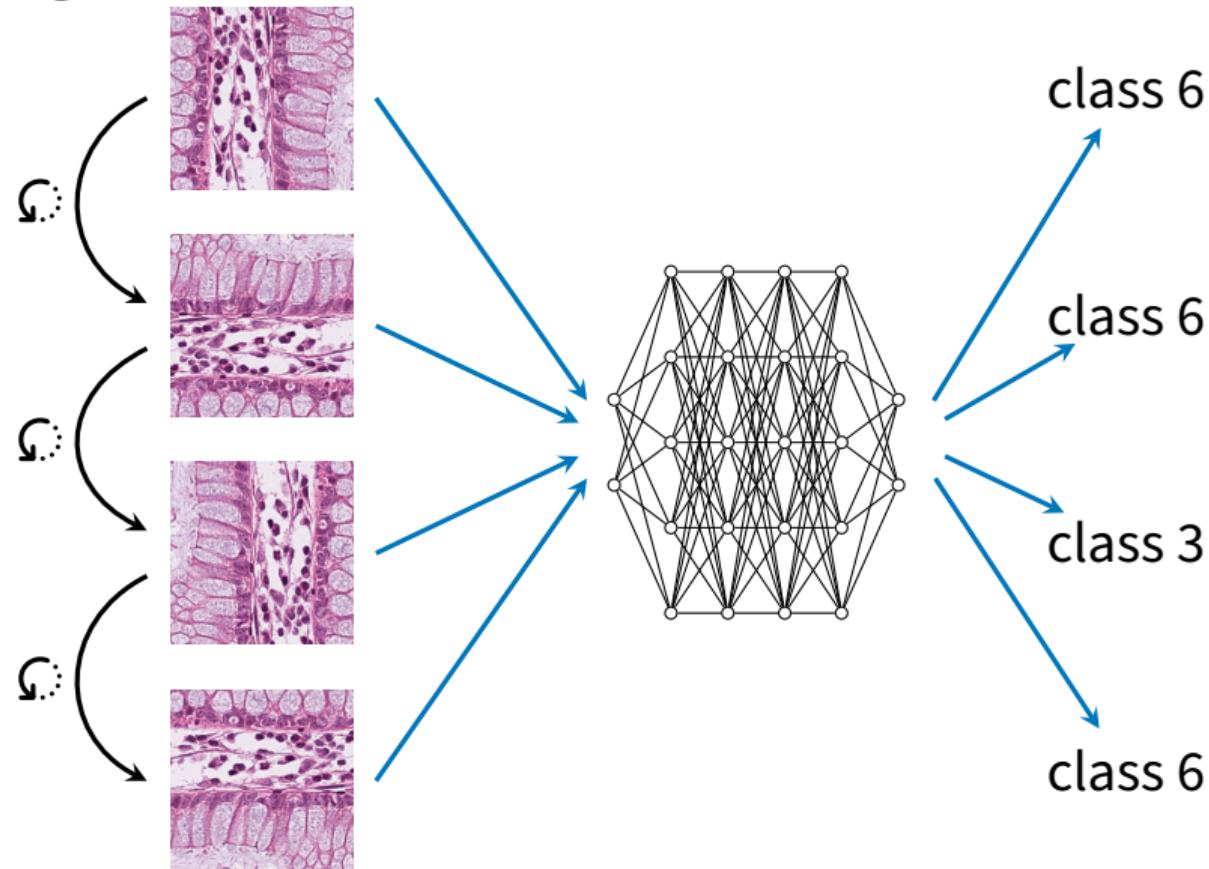


class 6

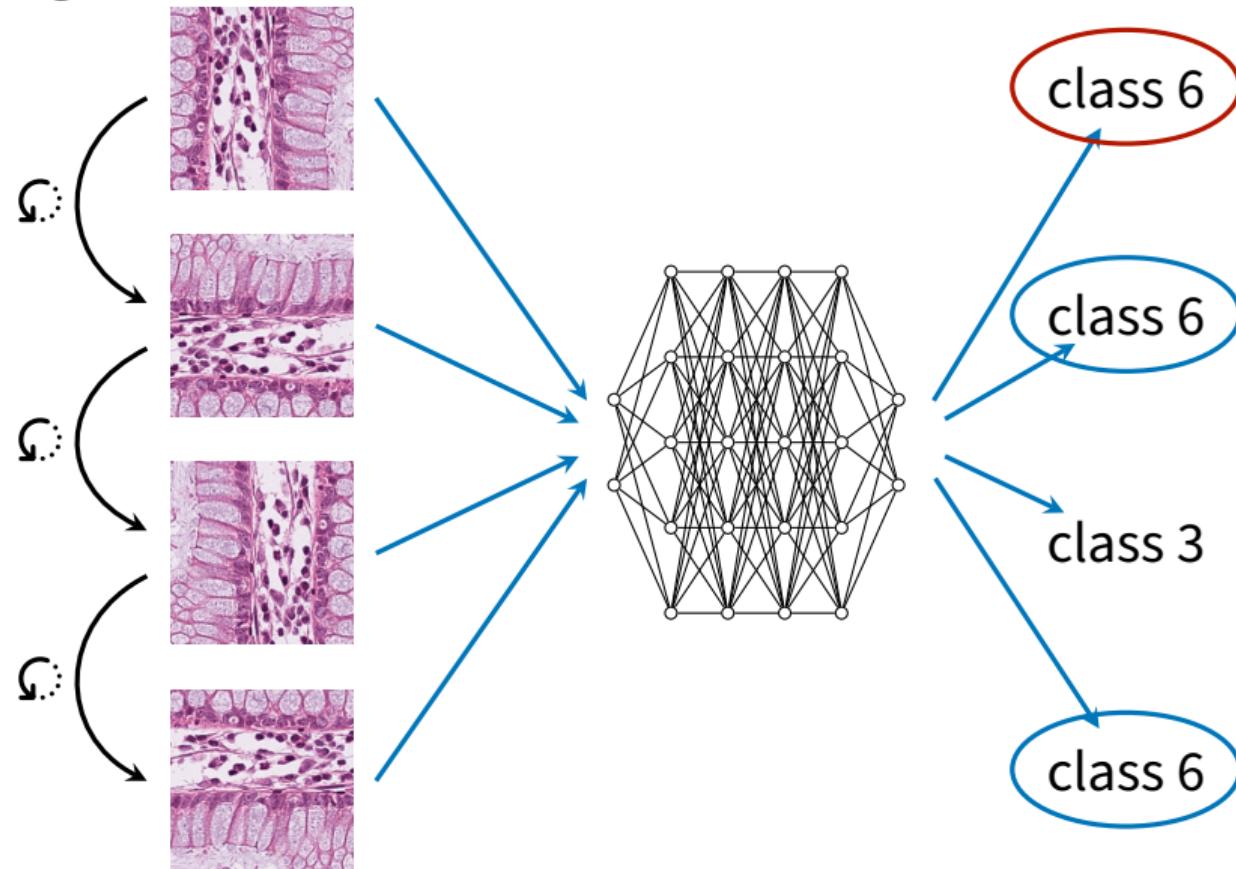
Histological slices



Histological slices

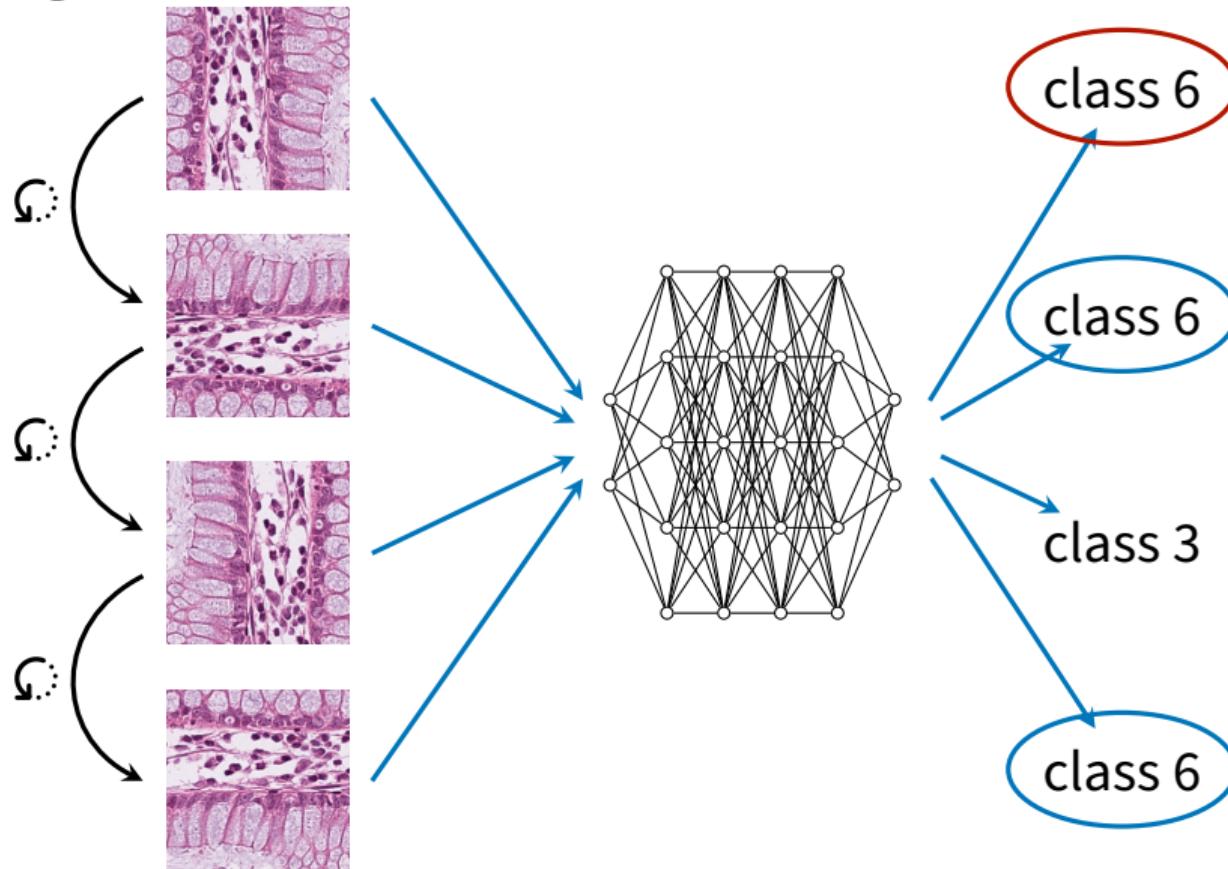


Histological slices

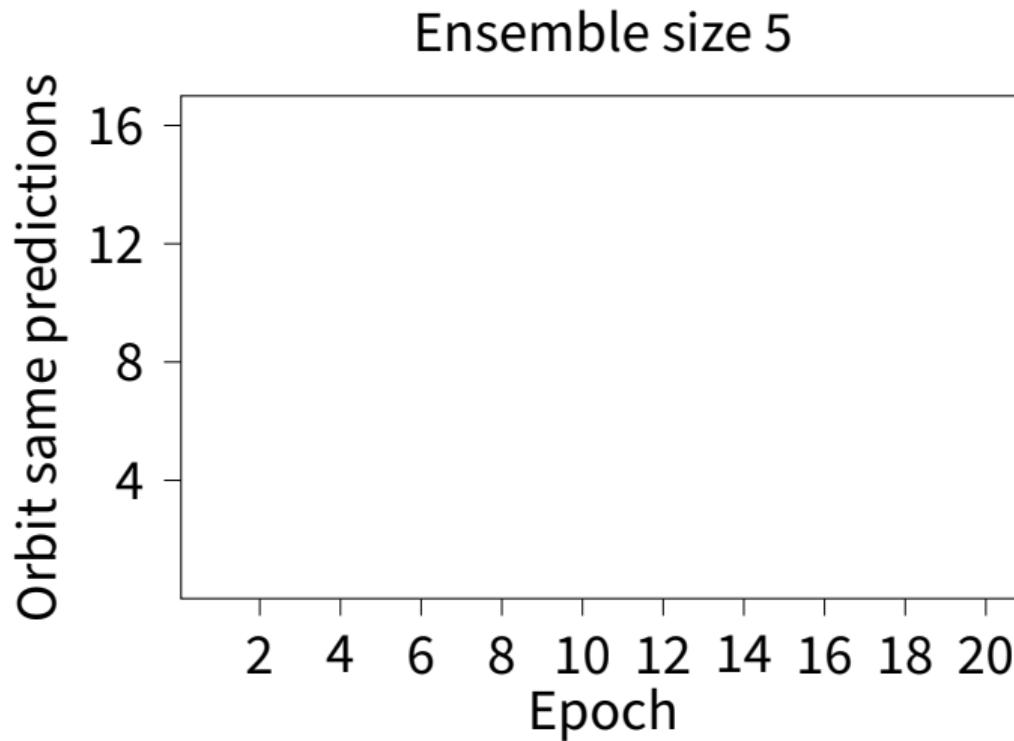


Histological slices

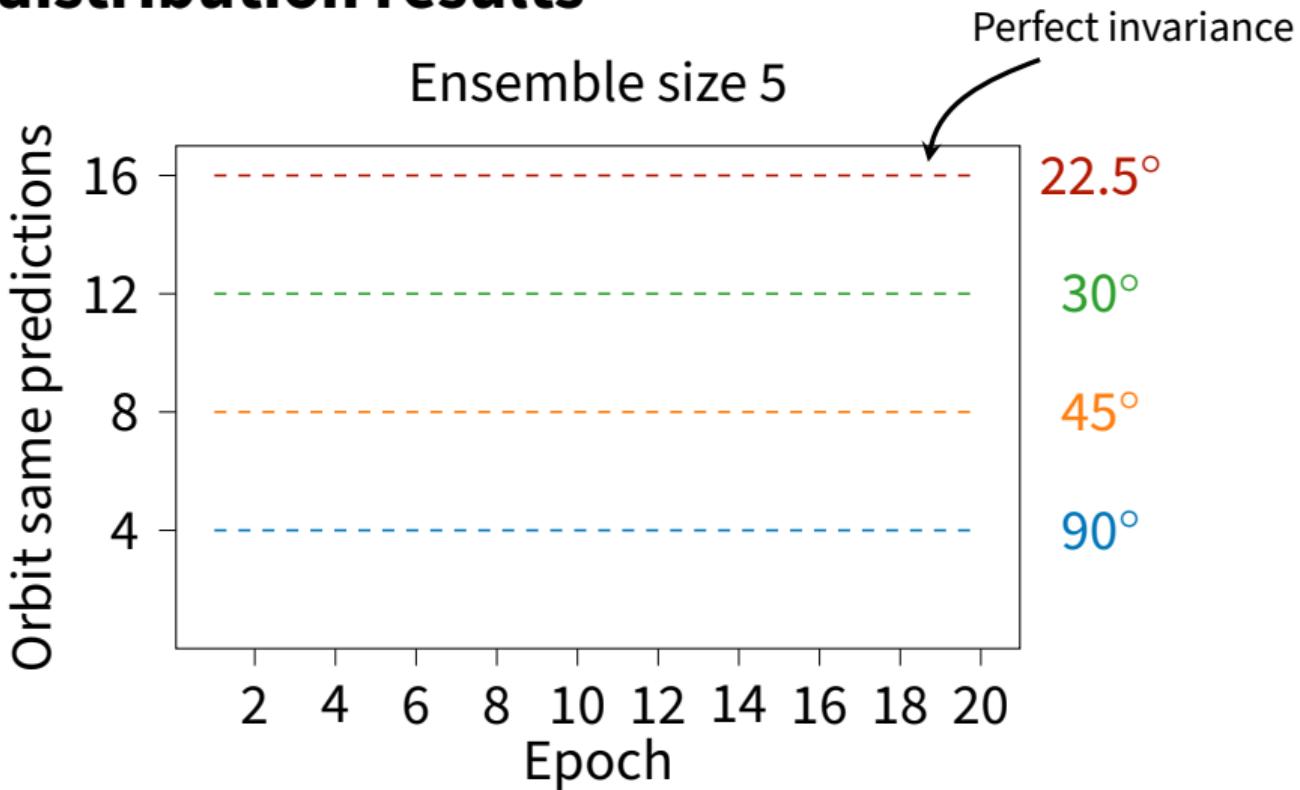
Orbit Same Predictions = 3



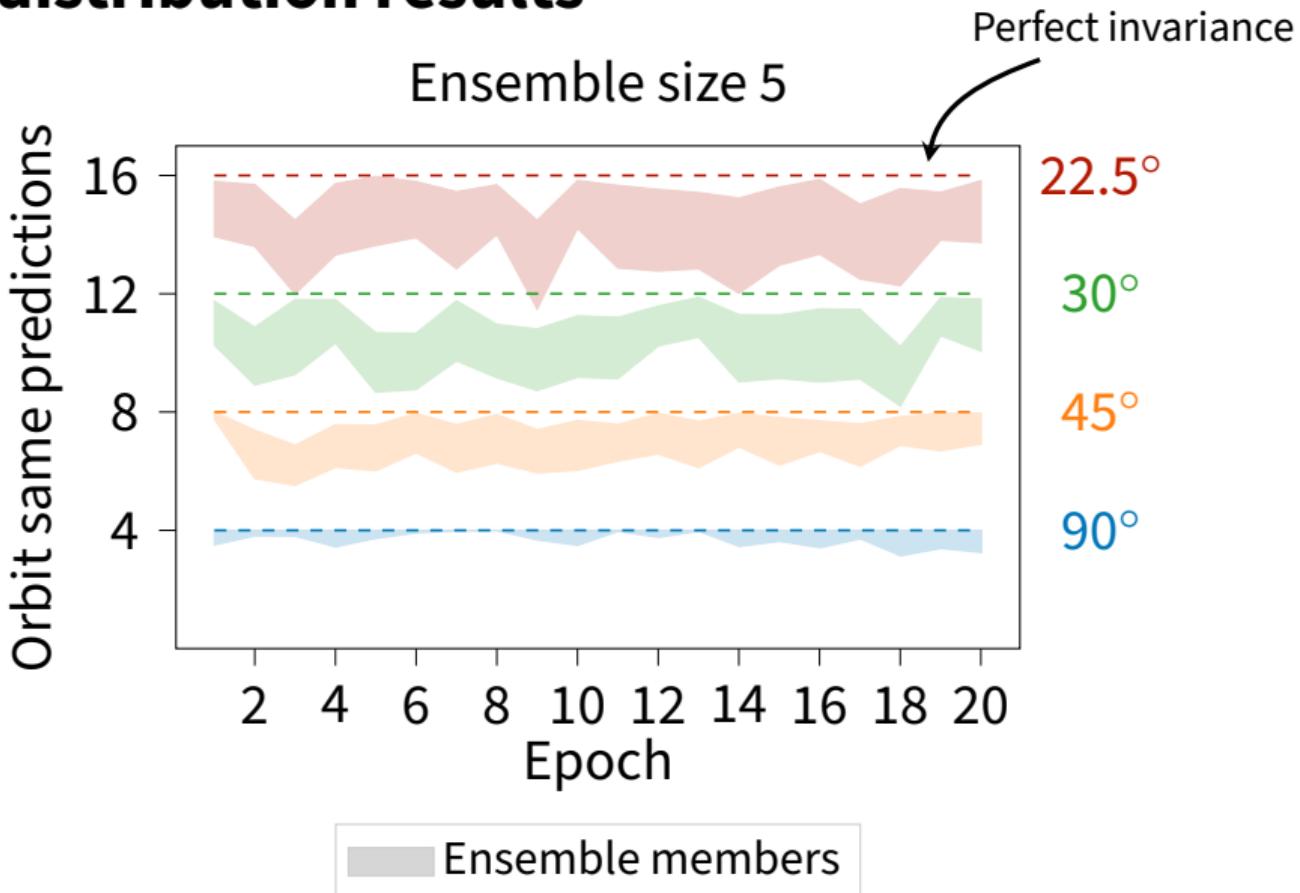
Out of distribution results



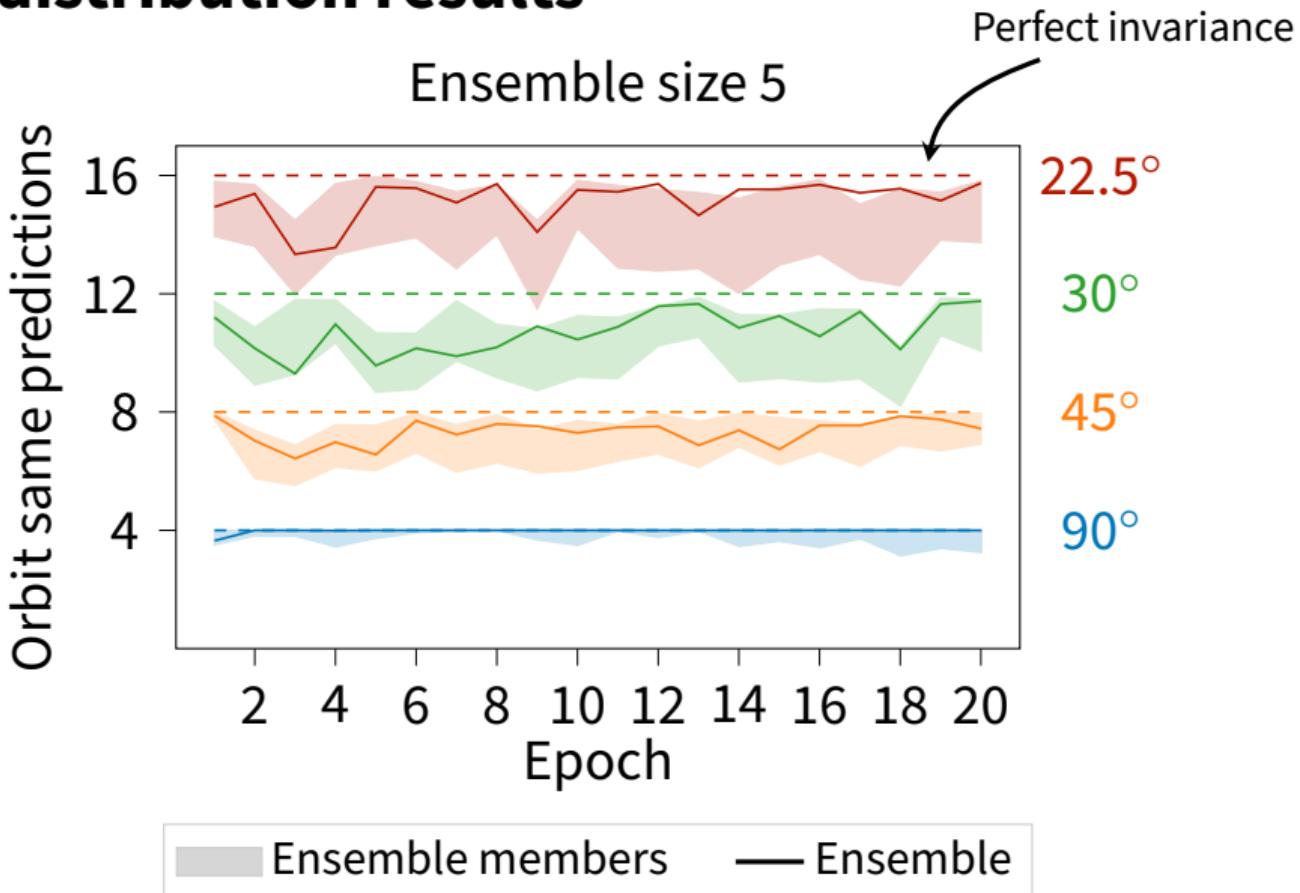
Out of distribution results



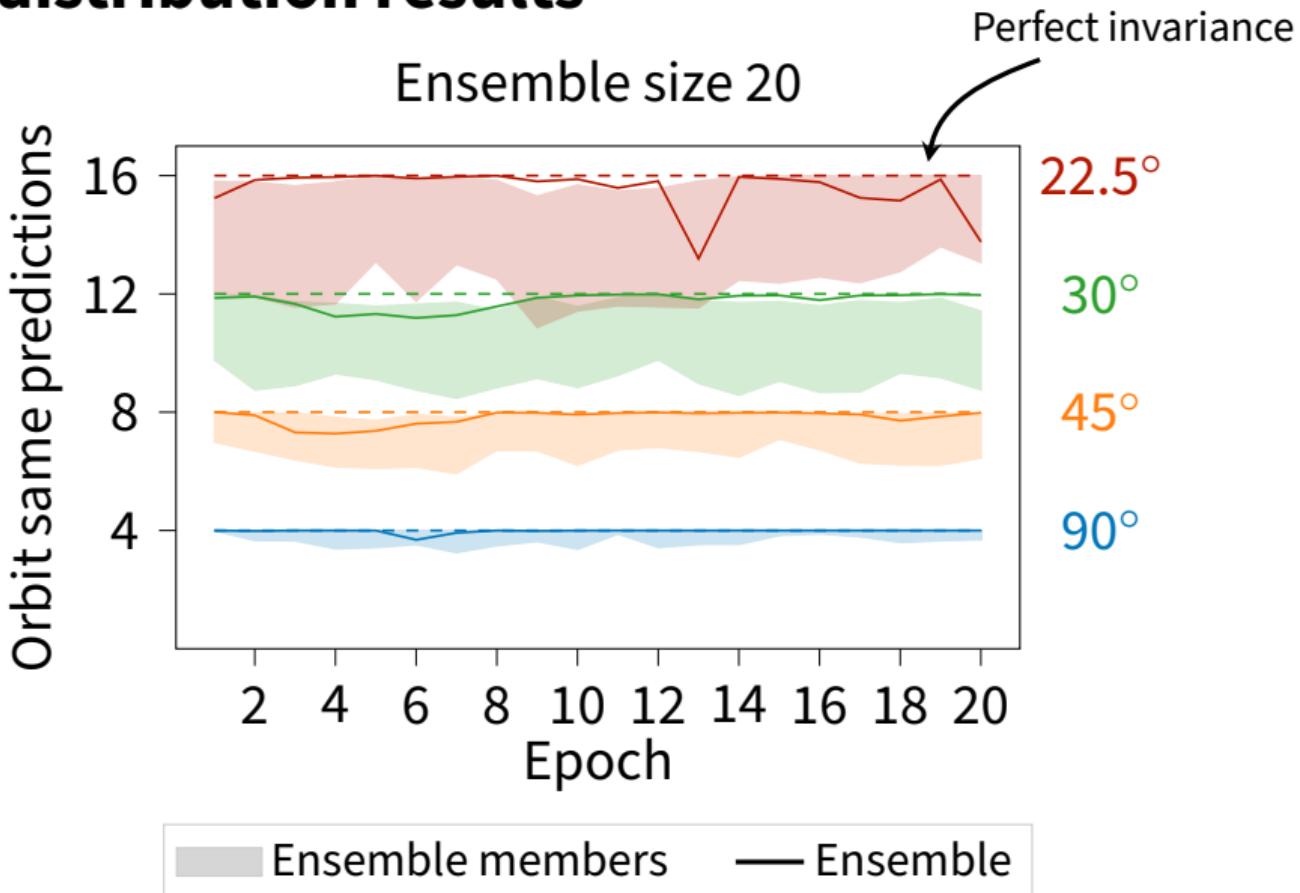
Out of distribution results



Out of distribution results



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

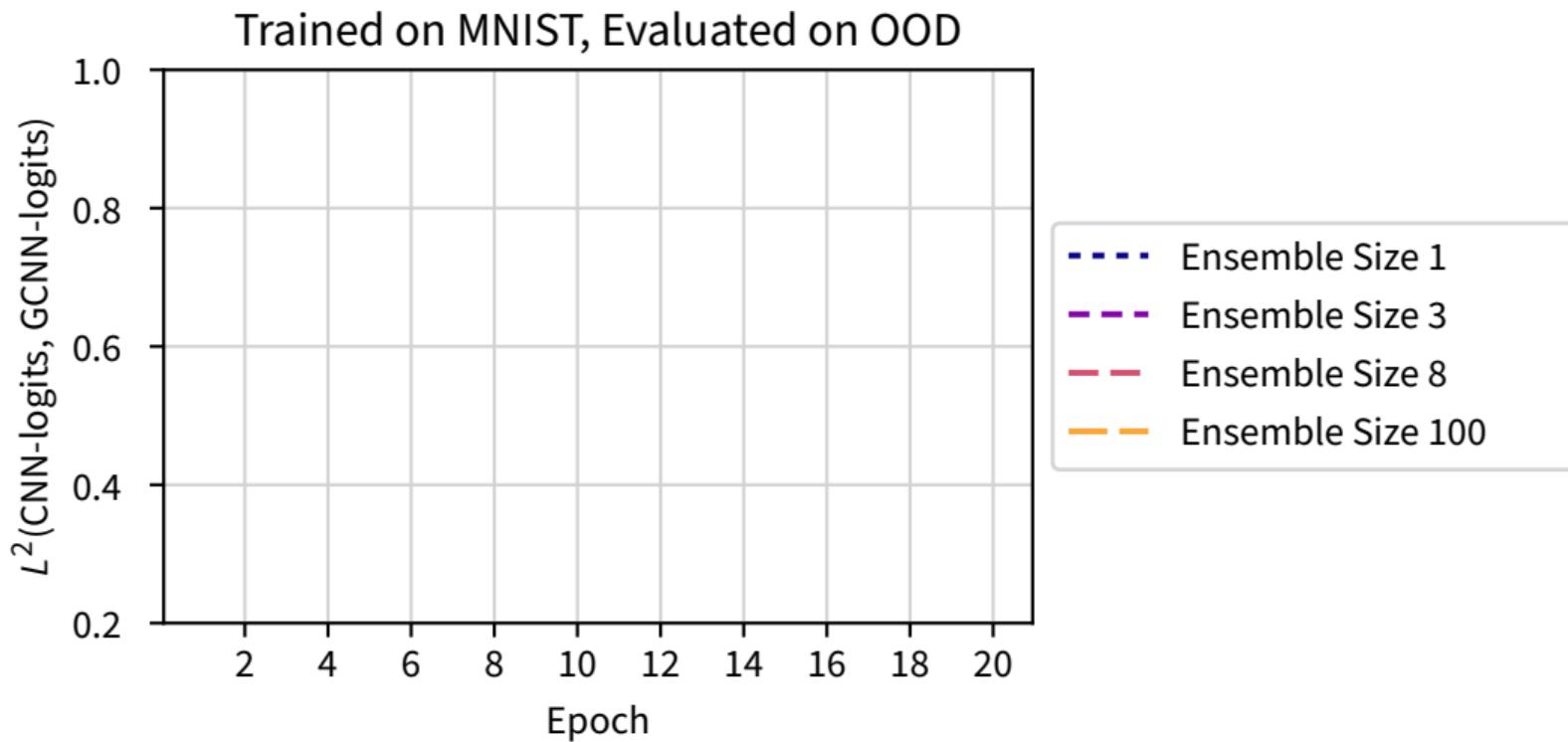
Orbit same predictions out of distribution:

	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

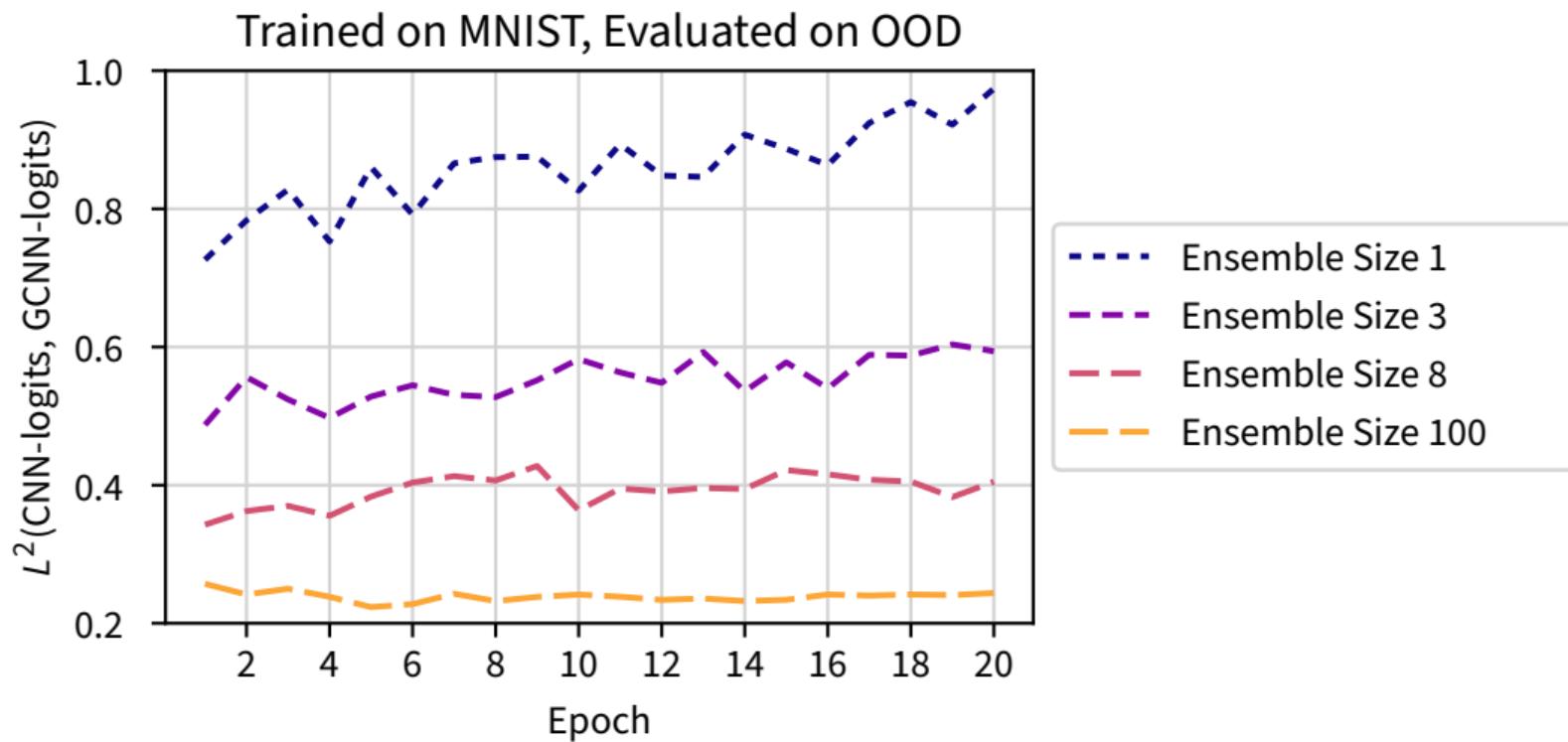
¹[Weiler et al. 2019], ²[Kaba et al. 2022]

Convergence of augmented CNNs to GCNNs

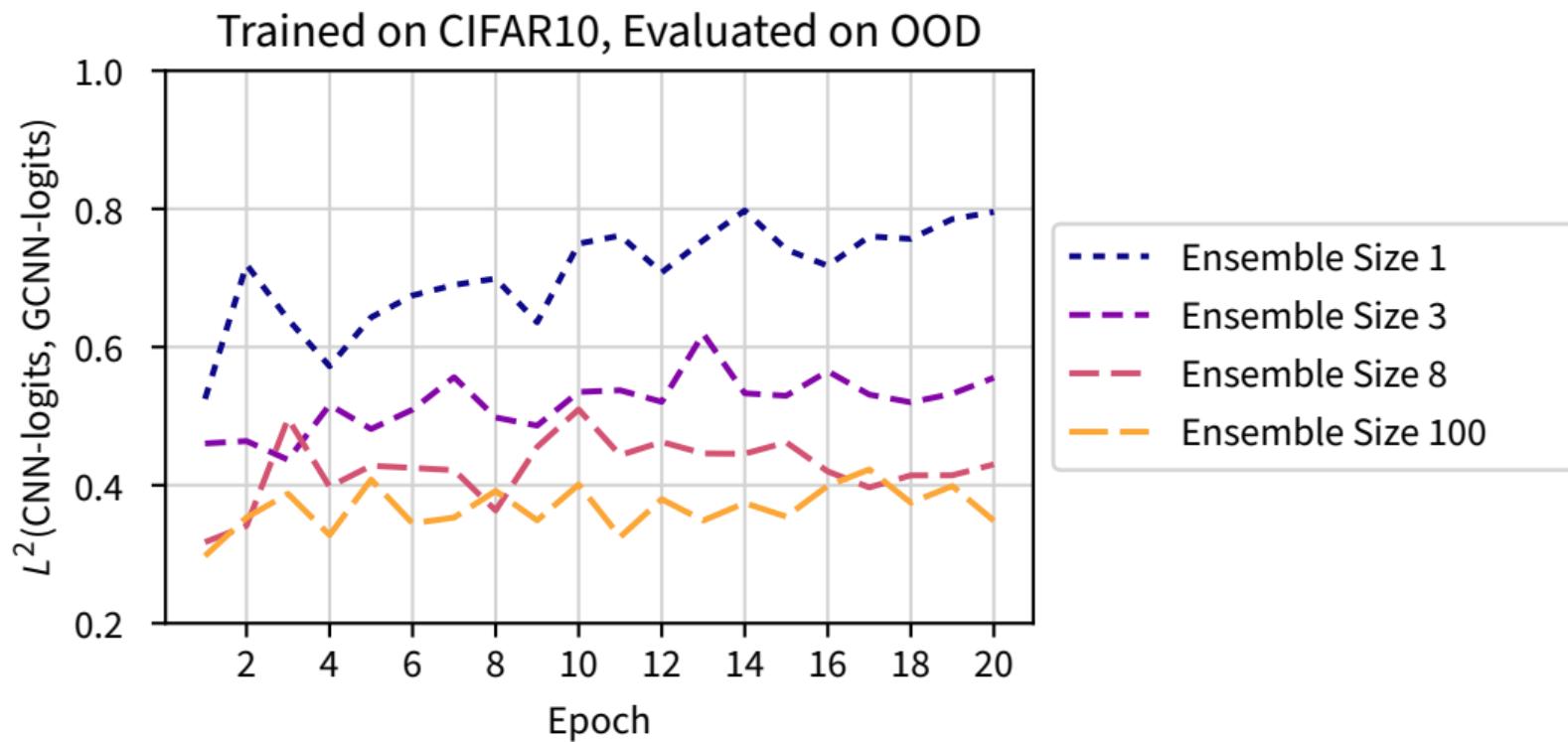
Convergence of augmented CNNs to GCNNs



Convergence of augmented CNNs to GCNNs



Convergence of augmented CNNs to GCNNs



Finite-width from Feynman diagrams

Restrictions of the infinite-width limit

The infinite-width limit...

- thumb-up Yields compact expressions
- thumb-up Is under complete analytic control

Restrictions of the infinite-width limit

The infinite-width limit...

- 👍 Yields compact expressions
- 👍 Is under complete analytic control
- 👎 Only Gaussian distributions
- 👎 Linearization of the model in the parameters
- 👎 No feature learning

Restrictions of the infinite-width limit

The infinite-width limit...

- 👍 Yields compact expressions
 - 👍 Is under complete analytic control
 - 👎 Only Gaussian distributions
 - 👎 Linearization of the model in the parameters
 - 👎 No feature learning
- ① Use methods from physics to compute finite-width effects

Field theory

- In field theory, consider probability distribution over fields (functions)
- Typically: Gaussian probability distributions with small corrections

Field theory

- In field theory, consider probability distribution over fields (functions)
- Typically: Gaussian probability distributions with small corrections
- The Gaussian limit corresponds to **free fields**
- Corrections correspond to interactions between fields

Field theory

- In field theory, consider probability distribution over fields (functions)
- Typically: Gaussian probability distributions with small corrections
- The Gaussian limit corresponds to **free fields**
- Corrections correspond to interactions between fields
- Compute statistics of these fields

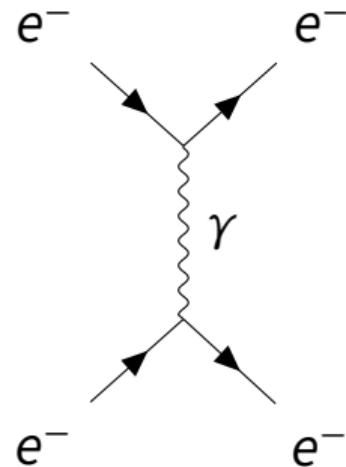
Feynman diagrams

Feynman diagrams are used to compute statistics

Feynman diagrams

Feynman diagrams are used to compute statistics

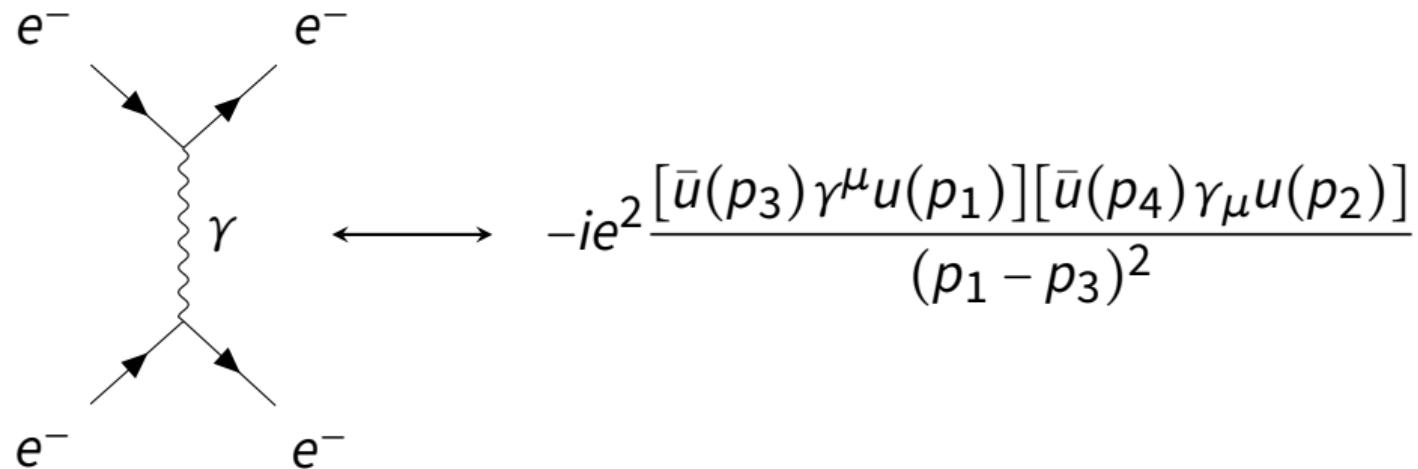
E.g. Electron-electron scattering



Feynman diagrams

Feynman diagrams are used to compute statistics

E.g. Electron-electron scattering



Non-Gaussian Corrections from Physics

- Taylor-expand network statistics in $1/\text{width}$
- Use Feynman diagrams to compute non-Gaussian corrections

Non-Gaussian Corrections from Physics

- Taylor-expand network statistics in 1/width
- Use Feynman diagrams to compute non-Gaussian corrections

Neural Networks	Field Theory
infinite width	no interactions
Gaussian distribution	free fields
finite-width	interactions

Tensors for neural network statistics

[Roberts, Yaida 2022]

Goal: Compute corrections for all neural network statistics at initialization, e.g.

Tensors for neural network statistics

[Roberts, Yaida 2022]

Goal: Compute corrections for all neural network statistics at initialization, e.g.

$$\mathbb{E}_{\theta}^c[z_{i_1}^{(\ell)}(x_1), z_{i_2}^{(\ell)}(x_2), \widehat{\Delta\Theta}_{i_3 i_4}^{(\ell)}(x_3, x_4)]$$

Tensors for neural network statistics

[Roberts, Yaida 2022]

Goal: Compute corrections for all neural network statistics at initialization, e.g.

$$\mathbb{E}_{\theta}^c[z_{i_1}^{(\ell)}(x_1), z_{i_2}^{(\ell)}(x_2), \widehat{\Delta\Theta}_{i_3 i_4}^{(\ell)}(x_3, x_4)]$$

Annotations:

- "cumulant" points to the first term \mathbb{E}_{θ}^c .
- "preactivation" points to the second term $z_{i_1}^{(\ell)}(x_1)$.
- "NTK fluctuation, $\widehat{\Delta\Theta}^{(\ell)} = \widehat{\Theta}^{(\ell)} - \mathbb{E}_{\theta}[\widehat{\Theta}^{(\ell)}]$ " points to the third term $\widehat{\Delta\Theta}_{i_3 i_4}^{(\ell)}(x_3, x_4)$.

Tensors for neural network statistics

[Roberts, Yaida 2022]

Goal: Compute corrections for all neural network statistics at initialization, e.g.

$$\mathbb{E}_{\theta}^c[z_{i_1}^{(\ell)}(x_1), z_{i_2}^{(\ell)}(x_2), \widehat{\Delta\Theta}_{i_3 i_4}^{(\ell)}(x_3, x_4)]$$

cumulant preactivation NTK fluctuation, $\widehat{\Delta\Theta}^{(\ell)} = \widehat{\Theta}^{(\ell)} - \mathbb{E}_{\theta}[\widehat{\Theta}^{(\ell)}]$

Decompose these into tensors

$$= \frac{1}{n} \left(D_{1234}^{(\ell)} \delta_{i_1 i_2} \delta_{i_3 i_4} + F_{1324}^{(\ell)} \delta_{i_1 i_3} \delta_{i_2 i_4} + F_{1423}^{(\ell)} \delta_{i_1 i_4} \delta_{i_2 i_3} \right)$$

Tensors for neural network statistics

[Roberts, Yaida 2022]

Goal: Compute corrections for all neural network statistics at initialization, e.g.

$$\mathbb{E}_{\theta}^c[z_{i_1}^{(\ell)}(x_1), z_{i_2}^{(\ell)}(x_2), \widehat{\Delta\Theta}_{i_3 i_4}^{(\ell)}(x_3, x_4)]$$

cumulant preactivation NTK fluctuation, $\widehat{\Delta\Theta}^{(\ell)} = \widehat{\Theta}^{(\ell)} - \mathbb{E}_{\theta}[\widehat{\Theta}^{(\ell)}]$

Decompose these into tensors

$$= \frac{1}{n} \left(D_{1234}^{(\ell)} \delta_{i_1 i_2} \delta_{i_3 i_4} + F_{1324}^{(\ell)} \delta_{i_1 i_3} \delta_{i_2 i_4} + F_{1423}^{(\ell)} \delta_{i_1 i_4} \delta_{i_2 i_3} \right)$$

Gram tensor, $F_{1324}^{(\ell)} = F^{(\ell)}(x_1, x_3, x_2, x_4)$

Tensors for neural network statistics

[Roberts, Yaida 2022]

⇒ Keeping track of these tensors is sufficient.

Tensors for neural network statistics

[Roberts, Yaida 2022]

⇒ Keeping track of these tensors is sufficient.

At order $1/n$, we have

Tensors for neural network statistics

[Roberts, Yaida 2022]

⇒ Keeping track of these tensors is sufficient.

At order $1/n$, we have

- $K, K^{\{1\}}$ and V_4 for preactivations

Tensors for neural network statistics

[Roberts, Yaida 2022]

⇒ Keeping track of these tensors is sufficient.

At order $1/n$, we have

- $K, K^{\{1\}}$ and V_4 for preactivations
- $\Theta, \Theta^{\{1\}}, A$ and B for derivatives

Tensors for neural network statistics

[Roberts, Yaida 2022]

⇒ Keeping track of these tensors is sufficient.

At order $1/n$, we have

- $K, K^{\{1\}}$ and V_4 for preactivations
- $\Theta, \Theta^{\{1\}}, A$ and B for derivatives
- D and F for mixed statistics

Tensors for neural network statistics

[Roberts, Yaida 2022]

⇒ Keeping track of these tensors is sufficient.

At order $1/n$, we have

- $K, K^{\{1\}}$ and V_4 for preactivations
- $\Theta, \Theta^{\{1\}}, A$ and B for derivatives
- D and F for mixed statistics
- 6 more for higher derivatives, needed for training

Tensor recursions

For each tensor, there is a layer-wise recursion relation, e.g.

Tensor recursions

For each tensor, there is a layer-wise recursion relation, e.g.

$$\begin{aligned} F_{1324}^{(\ell+1)} &= \langle \sigma_1^{(\ell)} \sigma_2^{(\ell)} \sigma_3'^{(\ell)} \sigma_4'^{(\ell)} \rangle_{K^{(\ell)}} \Theta_{34}^{(\ell)} \\ &+ \sum_{\alpha, \beta, \gamma, \delta=1}^4 \langle \sigma_1^{(\ell)} \sigma_3'^{(\ell)} z_\alpha^{(\ell)} \rangle_{K^{(\ell)}} \langle \sigma_2^{(\ell)} \sigma_4'^{(\ell)} z_\beta^{(\ell)} \rangle_{K^{(\ell)}} K_{(\ell)}^{\alpha \gamma} K_{(\ell)}^{\beta \delta} F_{\gamma 3 \delta 4}^{(\ell)} \end{aligned}$$

Tensor recursions

For each tensor, there is a layer-wise recursion relation, e.g.

$$F_{1324}^{(\ell+1)} = \langle \sigma_1^{(\ell)} \sigma_2^{(\ell)} \sigma_3'^{(\ell)} \sigma_4'^{(\ell)} \rangle_{K^{(\ell)}} \Theta_{34}^{(\ell)}$$

Gaussian expectation with cov. $K^{(\ell)}$

$$+ \sum_{\alpha, \beta, \gamma, \delta=1}^4 \langle \sigma_1^{(\ell)} \sigma_3'^{(\ell)} z_\alpha^{(\ell)} \rangle_{K^{(\ell)}} \langle \sigma_2^{(\ell)} \sigma_4'^{(\ell)} z_\beta^{(\ell)} \rangle_{K^{(\ell)}} K_{(\ell)}^{\alpha \gamma} K_{(\ell)}^{\beta \delta} F_{\gamma 3 \delta 4}^{(\ell)}$$

Components of $(K^{(\ell)})^{-1}$

Tensor recursions

For each tensor, there is a layer-wise recursion relation, e.g.

$$F_{1324}^{(\ell+1)} = \langle \sigma_1^{(\ell)} \sigma_2^{(\ell)} \sigma_3'^{(\ell)} \sigma_4'^{(\ell)} \rangle_{K^{(\ell)}} \Theta_{34}^{(\ell)}$$

Gaussian expectation with cov. $K^{(\ell)}$

$$+ \sum_{\alpha, \beta, \gamma, \delta=1}^4 \langle \sigma_1^{(\ell)} \sigma_3'^{(\ell)} z_\alpha^{(\ell)} \rangle_{K^{(\ell)}} \langle \sigma_2^{(\ell)} \sigma_4'^{(\ell)} z_\beta^{(\ell)} \rangle_{K^{(\ell)}} K_{(\ell)}^{\alpha \gamma} K_{(\ell)}^{\beta \delta} F_{\gamma 3 \delta 4}^{(\ell)}$$

Components of $(K^{(\ell)})^{-1}$

Solving this system of recursions yields the complete network statistics at order $1/n$.

Tensor recursions

For each tensor, there is a layer-wise recursion relation, e.g.

$$F_{1324}^{(\ell+1)} = \langle \sigma_1^{(\ell)} \sigma_2^{(\ell)} \sigma_3'^{(\ell)} \sigma_4'^{(\ell)} \rangle_{K^{(\ell)}} \Theta_{34}^{(\ell)}$$

Gaussian expectation with cov. $K^{(\ell)}$

$$+ \sum_{\alpha, \beta, \gamma, \delta=1}^4 \langle \sigma_1^{(\ell)} \sigma_3'^{(\ell)} z_\alpha^{(\ell)} \rangle_{K^{(\ell)}} \langle \sigma_2^{(\ell)} \sigma_4'^{(\ell)} z_\beta^{(\ell)} \rangle_{K^{(\ell)}} K_{(\ell)}^{\alpha \gamma} K_{(\ell)}^{\beta \delta} F_{\gamma 3 \delta 4}^{(\ell)}$$

Components of $(K^{(\ell)})^{-1}$

Solving this system of recursions yields the complete network statistics at order $1/n$.

⌚ Computing these recursions analytically is very laborious

Feynman diagrams

Use Feynman diagrams to compute these recursions.

Feynman diagrams

Use Feynman diagrams to compute these recursions.

- For preactivations, can read off Feynman rules from NN probability distribution, e.g.

[Banta et al. 2024]

$$z_\alpha \equiv \alpha \bullet \text{---} \quad \langle \quad \rangle_{K^{(\ell)}} \equiv \textcircled{\cdot} \quad \begin{array}{c} \beta \\ \swarrow \quad \searrow \\ a \end{array} \quad \widehat{\Delta G}_{i,\alpha\beta}^{(\ell)} \sim \frac{1}{n}$$

Feynman diagrams

Use Feynman diagrams to compute these recursions.

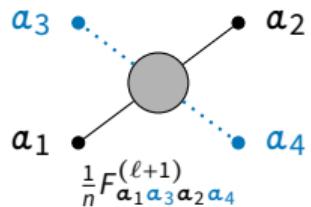
- For preactivations, can read off Feynman rules from NN probability distribution, e.g. [\[Banta et al. 2024\]](#)

$$z_\alpha \equiv \alpha \bullet \text{---} \quad \langle \quad \rangle_{K^{(\ell)}} \equiv \textcircled{\cdot} \quad \begin{array}{c} \beta \\ \swarrow \quad \searrow \\ a \end{array} \quad \widehat{\Delta G}_{i,\alpha\beta}^{(\ell)} \sim \frac{1}{n}$$

- For derivatives, need to find Feynman rules by inspecting analytic expressions

Feynman rules relevant for the F-tensor

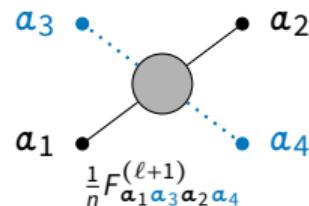
$$\widehat{\Delta\Theta}_{\alpha\beta} \equiv \begin{array}{c} \beta \\ \alpha \end{array} \cdot \dots \cdot \dots$$



$$\begin{array}{c} \beta \\ \alpha \end{array} \cdot \dots \cdot \sigma_{i,\alpha}^{(\ell)} \sigma_{i,\beta}^{'(\ell)} \sim \frac{1}{n}$$

Feynman rules relevant for the F-tensor

$$\widehat{\Delta\Theta}_{\alpha\beta} \equiv \begin{array}{c} \beta \\ \alpha \end{array} \cdot \dots \cdot \dots$$

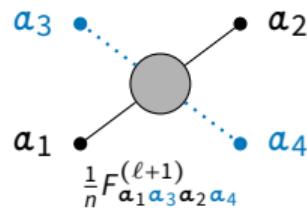


$$\begin{array}{c} \beta \\ \alpha \end{array} \cdot \dots \cdot \dots \sim \frac{1}{n}$$

The propagator \circlearrowleft satisfies selection rules, e.g.

Feynman rules relevant for the F-tensor

$$\widehat{\Delta\Theta}_{\alpha\beta} \equiv \begin{array}{c} \beta \\ \alpha \end{array} \cdot \dots \cdot \dots \cdot$$



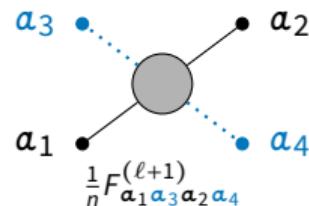
$$\begin{array}{c} \beta \\ \alpha \end{array} \cdot \dots \cdot \dots \cdot \sigma_{i,\alpha}^{(\ell)} \sigma_{i,\beta}^{'(\ell)} \sim \frac{1}{n}$$

The propagator satisfies selection rules, e.g.

- It cannot be directly connected to other propagators

Feynman rules relevant for the F-tensor

$$\widehat{\Delta\Theta}_{\alpha\beta} \equiv \begin{array}{c} \beta \\ \alpha \end{array} \cdot \dots \cdot \dots$$



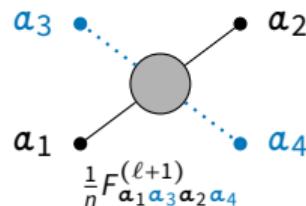
$$\begin{array}{c} \beta \\ \alpha \end{array} \cdot \dots \cdot \dots \sim \frac{1}{n}$$

The propagator satisfies selection rules, e.g.

- It cannot be directly connected to other propagators
- Dotted lines attached to a propagator do not appear in the Gaussian expectation value

Feynman rules relevant for the F-tensor

$$\widehat{\Delta\Theta}_{\alpha\beta} \equiv \beta_a \cdot \dots \cdot$$



A Feynman diagram showing a central gray circle. A solid black line labeled α connects to it from the bottom left. A dotted blue line labeled β connects to it from the top left. Two horizontal dashed blue lines, labeled $\sigma_{i,\alpha}^{(\ell)} \sigma_{i,\beta}^{'(\ell)}$, are attached to the circle. To the right of the diagram is the expression $\sim \frac{1}{n}$.

The propagator satisfies selection rules, e.g.

- It cannot be directly connected to other propagators
- Dotted lines attached to a propagator do not appear in the Gaussian expectation value
- Pairs of dashed lines of the same color connected to the propagator add a factor of Θ

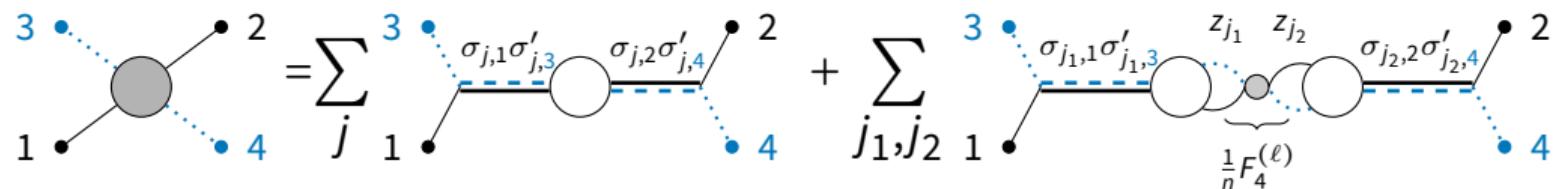
The F-recursion

Draw all diagrams possible for the F-tensor at order $1/n$

$$\text{Diagram with 4 external points (1, 2, 3, 4) connected to a central gray circle.} = \sum_j \text{Diagram with 4 external points (1, 2, 3, 4) connected to a central white circle via two horizontal lines labeled } \sigma_{j,1}\sigma'_{j,3} \text{ and } \sigma_{j,2}\sigma'_{j,4}. + \sum_{j_1, j_2} \text{Diagram with 4 external points (1, 2, 3, 4) connected to a central white circle via two horizontal lines labeled } \sigma_{j_1,1}\sigma'_{j_1,3} \text{ and } \sigma_{j_2,2}\sigma'_{j_2,4}. \text{ The central white circle is connected to a small gray circle labeled } z_{j_1} \text{ and } z_{j_2}, \text{ which is then connected to another white circle. A bracket below the second white circle is labeled } \frac{1}{n}F_4^{(\ell)}.$$

The F-recursion

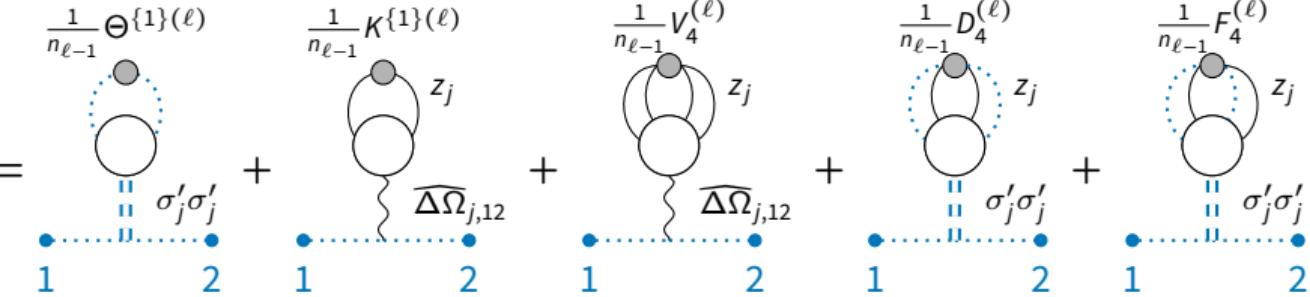
Draw all diagrams possible for the F-tensor at order $1/n$



Compare to the analytical expression

$$\begin{aligned} \frac{1}{n}F_{1324}^{(\ell+1)} &= \frac{1}{n}\langle\sigma_1^{(\ell)}\sigma_2^{(\ell)}\sigma_3'^{(\ell)}\sigma_4'^{(\ell)}\rangle_{K^{(\ell)}}\Theta_{34}^{(\ell)} \\ &+ \frac{1}{n}\sum_{\alpha,\beta,\gamma,\delta=1}^4\langle\sigma_1^{(\ell)}\sigma_3'^{(\ell)}z_\alpha^{(\ell)}\rangle_{K^{(\ell)}}\langle\sigma_2^{(\ell)}\sigma_4'^{(\ell)}z_\beta^{(\ell)}\rangle_{K^{(\ell)}}K_{(\ell)}^{\alpha\gamma}K_{(\ell)}^{\beta\delta}F_{\gamma3\delta4}^{(\ell)} \end{aligned}$$

The NTK recursion at finite width

$$\begin{aligned} \frac{1}{n_\ell} \Theta_{12}^{\{1\}(\ell+1)} &= \frac{1}{n_{\ell-1}} \Theta^{\{1\}(\ell)} + \frac{1}{n_{\ell-1}} K^{\{1\}(\ell)} z_j + \frac{1}{n_{\ell-1}} V_4^{(\ell)} z_j \\ &\quad + \frac{1}{n_{\ell-1}} D_4^{(\ell)} z_j + \frac{1}{n_{\ell-1}} F_4^{(\ell)} z_j \\ &\quad \text{with } \sigma'_j \sigma'_j = \Delta\bar{\Omega}_{j,12} \end{aligned}$$


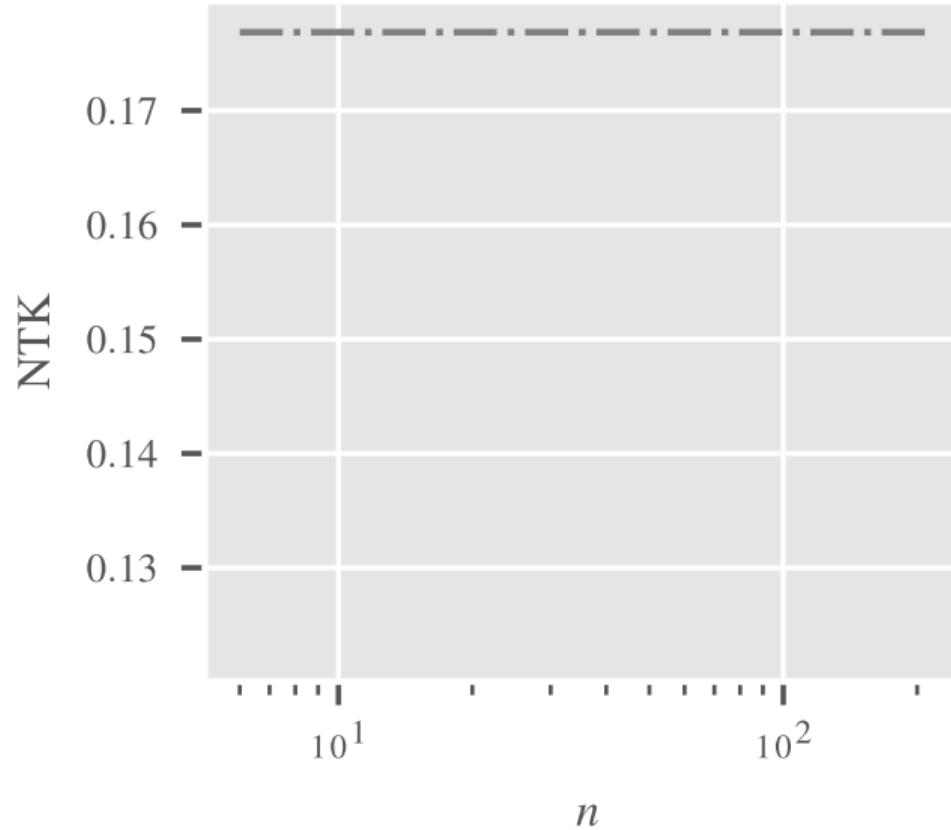
The NTK recursion at finite width

$$\begin{aligned} \frac{1}{n_{\ell-1}} \Theta^{\{1\}(\ell)}_{12} &= \frac{1}{n_{\ell-1}} \Theta^{\{1\}(\ell+1)}_{12} + \frac{1}{n_{\ell-1}} K^{\{1\}(\ell)} z_j \\ &\quad + \frac{1}{n_{\ell-1}} V_4^{(\ell)} z_j + \frac{1}{n_{\ell-1}} D_4^{(\ell)} z_j + \frac{1}{n_{\ell-1}} F_4^{(\ell)} z_j \end{aligned}$$

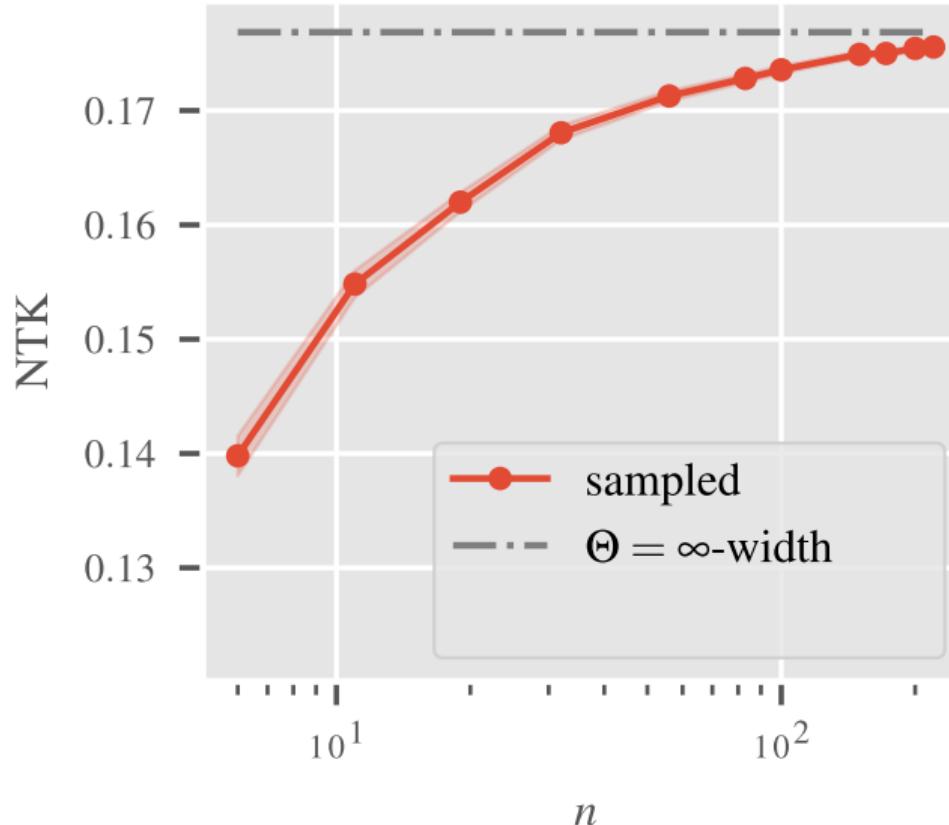
Diagram illustrating the NTK recursion at finite width. The left side shows a vertex with two external legs labeled 1 and 2, connected by a shaded circle. The right side is a sum of five terms, each with a shaded circle and a label above it. The first term is $\frac{1}{n_{\ell-1}} \Theta^{\{1\}(\ell)}_{12}$. The second term is $\frac{1}{n_{\ell-1}} K^{\{1\}(\ell)} z_j$, showing a shaded circle connected to a loop with two external legs labeled 1 and 2. The third term is $\frac{1}{n_{\ell-1}} V_4^{(\ell)} z_j$, showing a shaded circle connected to a loop with two external legs labeled 1 and 2, with a wavy line connecting the loop to the shaded circle. The fourth term is $\frac{1}{n_{\ell-1}} D_4^{(\ell)} z_j$, showing a shaded circle connected to a loop with two external legs labeled 1 and 2, with a dashed line connecting the loop to the shaded circle. The fifth term is $\frac{1}{n_{\ell-1}} F_4^{(\ell)} z_j$, showing a shaded circle connected to a loop with two external legs labeled 1 and 2, with a dashed line connecting the loop to the shaded circle.

! Feynman diagrams allow for much simpler derivation
of recursion relations

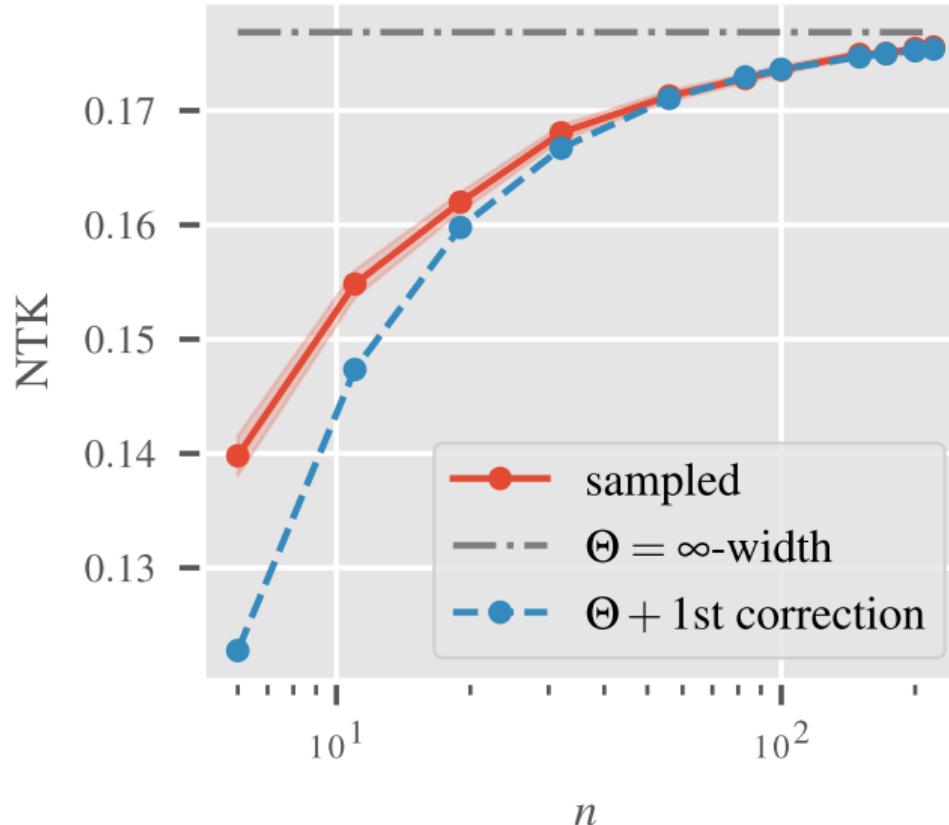
Numerical results



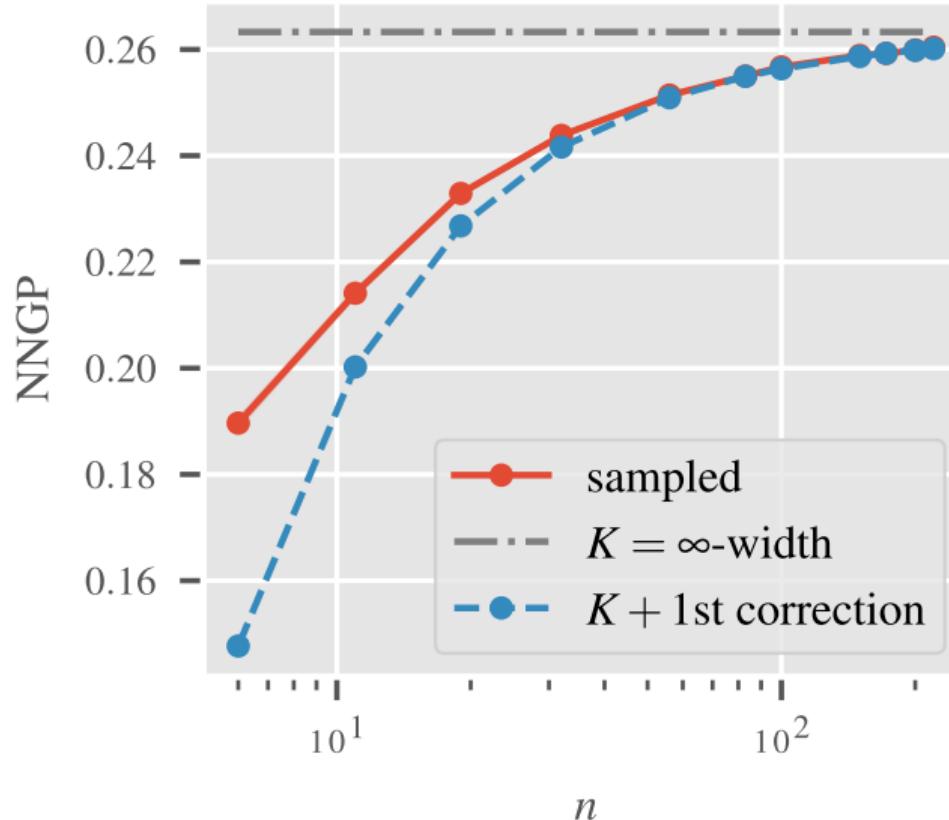
Numerical results



Numerical results



Numerical results



Key takeaways

Key takeaways

- Neural tangent kernels at infinite width can be used to understand data augmentation of deep ensembles:

Key takeaways

- Neural tangent kernels at infinite width can be used to understand data augmentation of deep ensembles:
 - Deep ensembles become exactly equivariant

Key takeaways

- Neural tangent kernels at infinite width can be used to understand data augmentation of deep ensembles:
 - Deep ensembles become exactly equivariant
 - Deep ensembles trained with data augmentation are group convolutional networks

Key takeaways

- Neural tangent kernels at infinite width can be used to understand data augmentation of deep ensembles:
 - Deep ensembles become exactly equivariant
 - Deep ensembles trained with data augmentation are group convolutional networks
- To consider non-Gaussian corrections away from infinite width, consider 1/width expansion

Key takeaways

- Neural tangent kernels at infinite width can be used to understand data augmentation of deep ensembles:
 - Deep ensembles become exactly equivariant
 - Deep ensembles trained with data augmentation are group convolutional networks
- To consider non-Gaussian corrections away from infinite width, consider 1/width expansion
- The corrections can be computed conveniently using Feynman diagrams

Papers

- Emergent Equivariance in Deep Ensembles
Jan E. Gerken*, Pan Kessel*
ICML 2024 (Oral)
 - Equivariant Neural Tangent Kernels
Philipp Misof, Pan Kessel, Jan E. Gerken
ICML 2025
 - Finite-Width Neural Tangent Kernels from Feynman Diagrams
Max Guillen*, Philipp Misof*, Jan E. Gerken
arXiv: 2508.11522
- * Equal contribution

Thank you!