MEASURING LEARNED EQUIVARIANCE: AN INFORMATION THEORETIC APPROACH

Anonymous authors
Paper under double-blind review

ABSTRACT

Learning symmetries through data augmentation offers an alternative to architectural equivariance, but does not guarantee that a network respects those symmetries. Measuring learned equivariance is especially difficult for intermediate representations, whose group actions are not specified. In a finite discrete setting, we show that equivariance requires representations of original and transformed inputs to determine one another exactly, yielding a test based on normalized mutual information (NMI). For high-dimensional activations, we use centered kernel alignment (CKA) as a scalable test, with an exact guarantee for orthogonal feature transformations. Across architectures and tasks, augmentation usually increases transformation consistency, showing that learned equivariance can be probed directly from representations without recovering the hidden transformation rules that relate them.

1 INTRODUCTION

Equivariance is a powerful inductive bias in fields such as biology, materials science, and image processing, letting networks produce equivalent outputs despite non-canonical inputs. However, recent work shows that at equal computational cost, non-equivariant networks can outperform structurally equivariant ones when they instead learn equivariance from a variety of transformed inputs (Abramson et al., 2024; Chai Discovery, 2024; Wohlwend et al., 2024). Such networks tend toward equivariance during training but are not constrained to it, so they may not be perfectly equivariant. Gruver et al. (2023) and Moskalev et al. (2023) found that models may have generalization error related to unlearned equivariance for inputs further from the training distribution. A practical measure of this error is therefore needed to check whether such models generalize correctly across inputs.

Prior work quantifying learned equivariance in model outputs and latent features assumes the group action on input, hidden, and output features is known and specified a priori (Gruver et al., 2023). This limits these metrics, since in practice the group action may be unknown (especially on the latent space) so checking for equivariance requires testing different candidate actions.

We address this with an information-theoretic approach. Maximal normalized mutual information between a layer's representations of an input and its transformed copy implies a deterministic, invertible map between them, as equivariance requires, without knowing what that map is. Because mutual information cannot be estimated reliably for continuous activations, we use centered kernel alignment (CKA) (Kornblith et al., 2019) as a proxy, and show that it is an exact test when the feature transformation is orthogonal.

Our contributions are:

- We show how learned equivariance can be tested without specifying how hidden features should transform: exact equivariance requires NMI to be maximal between original and transformed representations.
- We introduce CKA as a scalable geometric test for high-dimensional activations (comparing their pairwise similarity structure), show that exact equivariance yields maximal CKA under orthogonal feature transformations, and characterize where it departs from the information-theoretic criterion.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

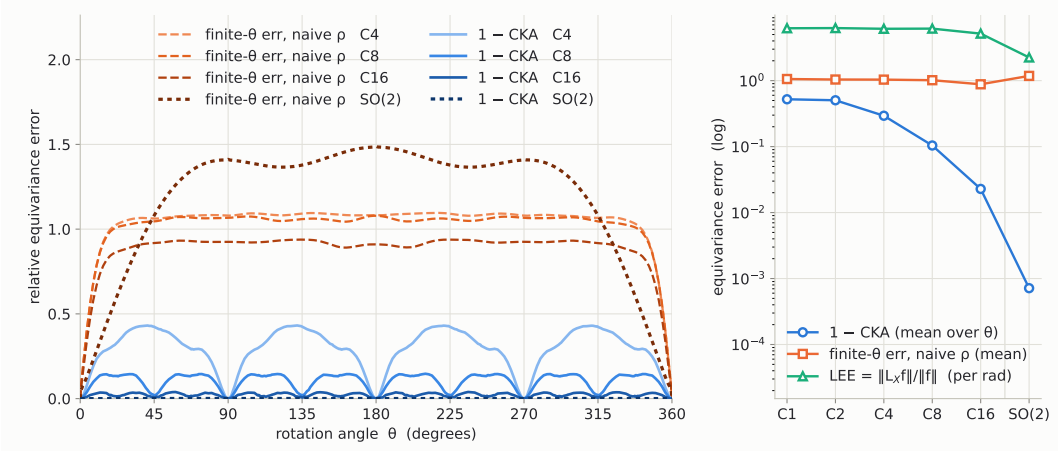

Figure 1: Comparison of equivariance error between $1-\text{CKA}$ and the local equivariance error (LEE) using the assumptions of Gruver et al. (2023). Each model is an exactly equivariant model made with ESCNN (Weiler & Cesa, 2019) and equivariance error is computed at every half a degree in the range $[0, 359.5]$. CKA is able to properly detect the zeros at the right intervals where the LEE is unable to detect equivariance for any model, even the SO(2).

- Across architectures, tasks, and symmetry groups, we show that data augmentation generally increases learned equivariance in standard networks, and identify where existing equivariance measures succeed or fail.

2 Related Work

Measuring and discovering equivariance. Lenc & Vedaldi (2015) measure equivariance by fitting feature-space maps for known image transformations, while Gruver et al. (2023) use Lie derivatives to measure local deviations from specified input and output actions. LieGG (Moskalev et al., 2022) recovers generators of learned invariances from a gradient-based matrix, while LieGAN (Yang et al., 2023) discovers distribution-preserving transformations through adversarial training. Our NMI and CKA measurements take the input transformations as given and compare activations on original and transformed inputs. They require neither specifying nor fitting the corresponding transformations of hidden features. SEIS (Lin et al., 2026) reports separate equivariance and invariance scores without fitting feature transformations; we show its equivariance score drops for steerable networks whose rotations also mix channels.

Information-theoretic symmetry. In a grid-world model of shared concept formation, Möller & Polani (2023) measure equivariance using mutual information between concepts assigned to original and transformed world states. Charvin et al. (2023) connect equivariance of discrete probabilistic mappings to optimal joint input-output compression that preserves mutual information. Our focus is measuring existing neural representations under given input transformations, rather than finding an optimal compression. For finite discrete representations, we express the necessary condition through NMI: original and transformed representations must determine one another exactly. We then use CKA to compare high-dimensional activations and investigate where it agrees with or departs from this condition.

Representation similarity. SVCCA (Raghu et al., 2017) compares correlated feature combinations after discarding low-variance directions, while CKA (Kornblith et al., 2019) compares patterns of similarity between examples. We use linear CKA across layers and during training because it requires neither selecting feature directions nor fitting an alignment. We also evaluate SVCCA and Procrustes error. Procrustes measures the mismatch remaining after the best rotation or reflection in feature space (Williams et al., 2021). We show that exact equivariance under orthogonal feature

transformations yields maximal CKA, connecting its established invariance properties to a necessary condition for equivariance.

Similarity scores need not track model behavior (Ding et al., 2021; Davari et al., 2022), and finite-sample bias can inflate CKA (Murphy et al., 2024). For equivariance, low CKA need not mean information was lost, and high CKA does not establish that a particular transformation rule was followed. We therefore use CKA to track transformation consistency across layers and training, with shuffled-pair nulls as a chance baseline and SVCCA and Procrustes as complementary checks.

Learning symmetry from data. Prior work studies how data, augmentation, and architecture affect learned symmetry (Bouchacourt et al., 2021; Bruintjes et al., 2023). Wang et al. (2023) analyze how mismatches between symmetry assumptions and the task or data affect prediction error. Whereas Bruintjes et al. (2023) match individual feature maps across channels, we compare whole representations with CKA. We use this comparison to examine which layers change with augmentation and how those changes develop during training, across image-recognition, image-to-image, and point-cloud tasks.

3 Background

We provide background on information theory (Shannon, 1948), equivariance, and CKA (Kornblith et al., 2019), including the Hilbert–Schmidt Independence Criterion (HSIC) (Gretton et al., 2007) on which CKA is built.

3.1 Information Theory

For a discrete random variable $X \sim p(x)$, the Shannon entropy $H(X) = -\sum_x p(x) \log p(x)$ quantifies the information gained by observing X. Given a second variable $Y \sim p(y)$, the conditional entropy is $H(X|Y) = -\sum_y \sum_x p(x, y) \log p(x|y)$. The mutual information between the pair of random variables is determined by:

$$I(X;Y) = H(X) - H(X|Y) \tag{1}$$

Mutual information is 0 if and only if X and Y are independent, i.e., observing Y yields no information about X.

As mutual information is dependent on the total available information, we utilize normalized mutual information (NMI):

$$\mathrm{NMI}(X,Y) = \frac{2I(X;Y)}{H(X) + H(Y)} \tag{2}$$

3.2 Equivariance

A function $f\colon \mathcal{X} \to \mathcal{Y}$ is G-equivariant for a group G if $f(\rho_x(g)x) = \rho_y(g)f(x)$ for all $g \in G$ and $x \in \mathcal{X}$, where ρ_x and ρ_y are the group actions on the input and output. Invariance is the special case $\rho_y(g) = \mathrm{id}$, giving $f(\rho_x(g)x) = f(x)$. As we work with images, we focus on the C_4 group of 90° rotations, which preserve an image's information.

There are a few ways to measure equivariance error, but one of the simplest ways is to compute the error between the output of the function on the transformed input and the output of the function transformed by the group action $\mathrm{EE} = ||f(\rho_{in}(g)x) - \rho_{out}(g)f(x)||_2^2$. Gruver et al. (2023) utilizes equivariance error to measure the per layer equivariance of a model given a infinitesimal transformation of the input. In practice, they take $\rho_{out}(g)$ to be trivial for scalar data or a purely spatial rotation for image-like data, which is wrong for steerable equivariant networks (Weiler & Cesa, 2019).

3.3 Hilbert-Schmidt Independence Criterion

Given two datasets $\mathbf{X} \in \mathbb{R}^{n \times d_\mathbf{X}}$ and $\mathbf{Y} \in \mathbb{R}^{n \times d_\mathbf{Y}}$ and feature maps ϕ and θ into a Hilbert space, we form kernels $K_\mathbf{X}(x, x') = \langle \phi(x), \phi(x') \rangle$ and $K_\mathbf{Y}(y, y') = \langle \theta(y), \theta(y') \rangle$ that capture nonlinear relationships. Note that we obtain a linear kernel by choosing θ and ϕ to be the identity function

and we obtain an RBF kernel by setting $K_{\mathbf{X}}$ to be $\exp(-||x - x'||^2/2\sigma^2)$. With centering matrix $H = \mathbf{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^\top$, the Hilbert-Schmidt Independence Criterion (HSIC) is defined as:

$$\text{HSIC}(K_{\mathbf{X}}, K_{\mathbf{Y}}) = \langle HK_{\mathbf{X}}H, HK_{\mathbf{Y}}H\rangle_F \tag{3}$$

3.4 Centered Kernel Alignment

Centered Kernel Alignment (CKA) normalizes HSIC to the range $[0, 1]$, removing dependence on the scale of the original data:

$$\text{CKA}(K_{\mathbf{X}}, K_{\mathbf{Y}}) = \frac{\text{HSIC}(K_{\mathbf{X}}, K_{\mathbf{Y}})}{\sqrt{\text{HSIC}(K_{\mathbf{X}}, K_{\mathbf{X}})\text{HSIC}(K_{\mathbf{Y}}, K_{\mathbf{Y}})}} \tag{4}$$

A value of 0 indicates no dependence captured by the kernel and 1 a perfect statistical relationship between $\mathbf{X}$ and $\mathbf{Y}$.

4 Methodology

We present MIME, a way to utilize mutual information to measure equivariance in discrete cases, and CKALE[1], a way to utilize CKA to measure equivariance in real models. We show theoretically that both can operate as an equivariance measure.

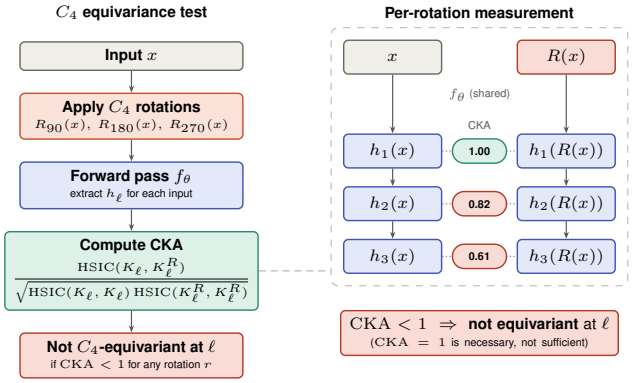

Figure 2: Per-layer CKALE test for C_4: CKA < 1 at any rotation rules out exact equivariance at that layer.

4.1 MIME: Mutual Information Measure of Equivariance

Let f_θ be a neural network and denote the truncated mapping to the output of layer ℓ as h_ℓ. For inputs X and group G, equivariance cannot be ruled out at layer ℓ if for all $g \in G$, $\text{NMI}(h_\ell(X), h_\ell(\rho(g)X)) = 1$. That is, the layer's outputs on the original and group-transformed data share all of their information, so a deterministic map relates $h_\ell(X)$ and $h_\ell(\rho(g)X)$ in both directions, as equivariance requires. We call this mutual information measure of equivariance MIME.

We show that NMI being 1 is a necessary but not sufficient condition for a function to be equivariant, which justifies its use as a measure.

Proposition 1. *Let $g \in G$. If $\text{NMI}(f(X), f(\rho(g)X)) < 1$, then f is not G-equivariant at g.*

Proof. Proof by contrapositive. Let X be a random variable taking values in $\mathcal{X}$ and let $f\colon \mathcal{X} \to \mathcal{Y}$ be G-equivariant, so that $f(\rho_x(g)x) = \rho_y(g)f(x)$ for all $x \in \mathcal{X}$. Define the random variables $Y := f(X)$, and $Y' := f(\rho_x(g)X)$. By equivariance, $Y' = \rho_y(g)Y$, and since $\rho_y(g)$ is a group action and hence invertible, $Y = \rho_y(g)^{-1}Y'$. Each of Y and Y' is therefore a deterministic function of the other, so $H(Y \mid Y') = H(Y' \mid Y) = 0$, and by Lemma 1, $\text{NMI}(Y, Y') = \text{NMI}(f(X), f(\rho_x(g)X)) = 1$. □

4.2 CKALE: CKA for Learned Equivariance

Mutual information requires the full data distribution, which is intractable for a layer's continuous, high-dimensional outputs: recovering the density on $\mathbb{R}^d$ would require evaluating every possible

[1]Pronounce like kale

Table 1: NMI between filtered original and rotated images for the Sobel-X filter (not C_4-equivariant) and the Laplacian (C_4-equivariant) on a restricted, discretized image set. The Laplacian stays at 1. Sobel-X is C_2-equivariant and anti-commutes, so it reads 1 at $180°$; a metric assuming the output action matches the input action would report a large error here, but NMI detects the equivariance without being told the action is a sign flip.

	Filter	
Rotation	Sobel-X	Laplacian
$90°$	**0.1954**	1.0000
$180°$	1.0000	1.0000
$270°$	**0.1954**	1.0000

input. Instead, to measure learned equivariance in real models, we apply CKA to each layer's output and its rotated copy to test whether the layer acts in an equivariant manner (Figure 2), a method we call CKALE: CKA for Learned Equivariance. We measure $\mathrm{CKA}(h_\ell(x), h_\ell(\rho_{in}(g)x))$ for all ℓ in the network f_θ. For any discrete group, we test every possible input operation when measuring equivariance, but for infinite groups such as $SO(2)$ we subsample to 16 samples to test for equivariance. We show that similar to NMI, CKA being 1 is a necessary but not sufficient condition for a function to be equivariant which justifies its use as a measure of equivariance.

Proposition 2. *Let $g \in G$. Assume $\rho_{\mathbf{Y}}(g)$ is orthogonal. Also assume orthogonally invariant kernels K that use the same feature map. (For example, linear or RBF kernels). If $\mathrm{CKA}(K_{f(\mathbf{X})}, K_{f(\rho_{\mathbf{X}}(g)\mathbf{X})}) < 1$, the function f is not G-equivariant at g.*

Proof. Proof by contrapositive. Assume $f\colon \mathbf{X} \to \mathbf{Y}$ is G-equivariant and $\mathbf{X}$ and $\mathbf{Y}$ are sets. Let $\mathbf{Y} := f(\mathbf{X})$ and $\mathbf{Y}' := f(\rho_{\mathbf{X}}(g)\mathbf{X})$. Since f is G-equivariant, there exists an invertible orthogonal map $\rho_{\mathbf{Y}}(g)$ such that $\mathbf{Y}' = \rho_{\mathbf{Y}}(g)\mathbf{Y}$. Following our two assumptions, $\rho_{\mathbf{Y}}(g)$ is orthogonal and our kernel functions $K_{\mathbf{Y}}$ and $K_{\mathbf{Y}'}$ are orthogonally invariant. Therefore $K_{\mathbf{Y}} = \langle\theta(y), \theta(y')\rangle = \langle\theta(\rho_y(g)y), \theta(\rho_y(g)y')\rangle = K_{\mathbf{Y}'}$. By Lemma 2, $\mathrm{CKA}(K_{\mathbf{Y}}, K_{\mathbf{Y}'}) = 1$. □

5 Experiments

We first validate MI as an equivariance measure and compare it with CKA, then show where MI breaks down, and finally compare CKALE with existing measures on real models.

5.1 Mutual Information as an Equivariance Measure

To show that mutual information can measure equivariance, we use a toy dataset of 4×4 binary images taking values in $\{-1, 1\}$. Only $2^{16} = 65536$ such images exist, so we run every possible image through the chosen filter and use the output rates to compute the probabilities and normalized mutual information. Outputs are binarized at a shared median and coarse-grained by summing each 2×2 quadrant. This commutes with the group action, since a $90°$ rotation permutes the quadrants and each 2×2 sum is rotation-invariant; Section 5.3 explains why it is needed.

As Table 1 shows, NMI falls below 1 only for Sobel-X at $90°$ and $270°$, since it is only C_2-equivariant.

Table 2 shows that relabeling the alphabet, a lossless bijection, drives CKA toward its null because CKA depends on geometry, while MI correctly stays at 1.000; both correctly flag the lossy merge. The null column shuffles the pairing between outputs. MI's high null reflects the large alphabet and makes it less trustworthy here, so CKA remains the more practical measure despite missing this form of equivariance.

5.2 CKA for Symmetry Probing

Mutual information is invariant to any bijective reparameterization of the features and falls only when information is genuinely destroyed. Linear CKA is invariant to a strictly smaller set of orthogonal transformations and isotropic scaling. Within that set the two agree exactly: on the identity and

Table 2: CKA versus MI on a discrete set of data points. We apply the Laplacian to the input image, then either keep the output unchanged, scramble the alphabet (relabel outputs while keeping points with the same label together), or merge the alphabet (map multiple distinct outputs to the same value). CKA reads the lossless relabeling as non-equivariant, while MI correctly detects equivariance. A high null indicates the metric may be less informative due to a large alphabet.

		linear CKA		exact MI		
readout	nature	value	null	bits	MI$/H$	null$/H$
identity	reference	1.0000	0.0004	7.3205	1.000	0.111
scramble p=1.00	bijective, **lossless**	0.0376	0.0009	**7.3205**	**1.000**	0.111
merge q=0.10	**90% of alphabet destroyed**	**0.0431**	0.0006	**3.1574**	**0.606**	0.005
Sobel-x control (not C4-equivariant)		0.0021	0.0004	1.2676	0.195	0.031

Table 3: Measuring equivariance after a Laplace operation is applied to the discretized images. The transformation is applied so we have a more interesting probability distribution under transformations. Under certain operations applied to the output we see how CKA, MI, and SVCCA either fails or succeeds to detect equivariance. SVCCA is able to detect one more group than CKA, but lacks an unbiased estimator, so we chose to utilize CKA.

readout	nature	MI	null	MI$/H$	linCKA	rbfCKA	SVCCA
identity	reference	7.3205	0.8113	1.000	1.0000	1.0000	1.0000
bit-permute	bijective, isometry	7.3205	0.8113	1.000	1.0000	1.0000	1.0000
ill-cond. linear	bijective, linear, *not* isometry	7.3205	0.8113	1.000	0.3584	0.2280	**1.0000**
alphabet-permute	bijective, *not* linear	7.3205	0.8113	1.000	0.1113	0.1543	**0.2597**
lossy-half	**many-to-one, destroyed**	**1.8643**	0.0065	**0.505**	0.4835	0.4528	0.5042

bit-permutation readouts of Table 3, both of which are orthogonal, CKA reads 1.0000 and MI reads 7.3205 bits at MI$/H = 1.000$. Outside it, CKA collapses while MI does not.

Every representation of a compact group is equivalent to an orthogonal one, so the $\rho(g)$ of an architecturally equivariant network can be taken orthogonal, which is CKA's invariance class. We assume this holds approximately for networks trained with augmentation. Because each comparison is between augmented and baseline models of identical architecture, systematic departures largely cancel. SVCCA is invariant to all invertible linear maps and survives the ill-conditioned readout, but canonical correlations are biased upward as feature dimension approaches sample count, which is the regime of our batch-size-64 sweep. HSIC admits an unbiased estimator (Song et al., 2012), which we use to compute CKA throughout (Nguyen et al., 2021).

Squared magnitudes are common in real networks (fiber norms in steerable CNNs, attention scores, second-moment features), and they break linear transformation behavior: if $v(gx) = Pv(x)$, the energy block $(NPv)^2$ contains cross terms $v_j v_k$, so no matrix $\rho(g)$ exists to assume or fit. We test this with the injective readout $\sigma(v) = [Mv \mid (Nv)^2]$. In Table 4, MI$/H$ stays at 1.000, so any reported error belongs to the metric rather than the representation. LEE sits at its shuffled null ($1.02\times$ separation) and detects nothing. The fitted maps still rank the kernels correctly, but they leave residuals near 0.7 on an exactly equivariant representation, which is a calibration failure that would read as non-equivariance in practice. CKA and MI both keep working, with linear CKA at 0.7379 against 0.0071 on the control.

5.3 Mutual Information Precision Problem

Higher precision enlarges the alphabet, and the shuffled null rises with it. Our earlier MI experiments used coarse-graining to keep the null low, but choosing a granularity is difficult in practice, so we repeat them at full 16-bit precision.

Table 5 shows that by increasing to the full 16-bit resolution, the trustworthiness of the MI measure falls because the null measurement rises.

Table 4: Comparison between different methods on detecting equivariance given an identity and an energy readout. All methods except LEE succeed on the identity readout. Under the energy readout, only mutual information stays exact, while CKA still clearly separates the equivariant kernel from the control.

		Laplace		Sobel-X		
	metric	value	null	value	null	separation
(a) identity readout — reference						
assumes ρ	LEE, $\rho = I \downarrow$	1.4933	1.4145	1.3959	1.4150	**0.93**×
fits ρ	affine fit ↓	**0.0000**	1.0001	1.0836	1.0001	∞
	Procrustes ↓	**0.0000**	1.4874	1.4884	1.4641	∞
no ρ	linear CKA ↑	1.0000	0.0020	0.0023	0.0007	435×
	RBF CKA ↑	1.0000	0.0030	0.0044	0.0014	227×
	SVCCA ↑	1.0000	0.0387	0.0936	0.0283	10.7×
	MI, exact (bits) ↑	7.3205	0.8100	1.2676	0.2027	5.8×
	MI/H (null/H)	1.000	0.111	0.195	0.031	—
(b) ENERGY readout $\sigma(v) = [Mv \mid (Nv)^2]$						
assumes ρ	LEE, $\rho = I \downarrow$	1.3640	1.4140	1.3958	1.4139	**1.02**×
fits ρ	affine fit ↓	0.6723	1.0002	1.0438	1.0002	1.55×
	Procrustes ↓	0.7225	1.4749	1.4511	1.4422	2.01×
no ρ	linear CKA ↑	0.7379	0.0029	0.0071	0.0015	**104**×
	RBF CKA ↑	0.8634	0.0042	0.0091	0.0023	**95**×
	SVCCA ↑	0.7802	0.0459	0.0876	0.0394	8.9×
	MI, exact (bits) ↑	**7.3205**	0.8100	1.2676	0.2027	5.8×
	MI/H (null/H)	**1.000**	0.111	0.195	0.031	—

Table 5: Changing the granularity of what mutual information sees from 4 bits to the full 16 bits shows that the null reading rises with it due to the larger alphabet, making mutual information a much worse measure the more unique outputs there are. In a continuous space with infinite outputs, mutual information is not only intractable to compute, but also loses its ability to distinguish real equivariance from scrambles.

granularity	symbols	samp./sym	MI	H	MI/H	null	margin
no convolution — raw pixels							
full 16-bit	65,536	1.00	16.0000	16.0000	1.000	16.0000	**0.0000**
Laplacian — exactly C4-equivariant							
full 16-bit	45,707	1.43	15.1834	15.1834	1.000	**14.3668**	**0.8165**
quadrant	486	134.8	7.3205	7.3205	1.000	0.8157	**6.5048**
Sobel-x — not C4-equivariant (control)							
full 16-bit	1,225	53.5	4.6122	9.0427	0.510	2.6101	**2.0020**
quadrant	139	471.5	1.2676	6.4871	0.195	0.2035	1.0641

5.4 CKALE AS AN ASSUMPTION FREE EQUIVARIANCE MEASURE

Other measures require assumptions about the group action. CKA's assumptions (orthogonal transformations or isotropic scaling) are much weaker. We compare measures on exactly equivariant models and on models trained with and without augmentation.

Table 6 and Fig. 1 show that LEE's assumed output action leads it to report high error where there should be zeros, while CKA and LieGG identify the correct behavior.

However, LieGG assumes an invariant output, so it misses equivariance in a model with vector outputs (Table 7).

Table 6: Three measures on exactly-SO(2) point clouds (three seeds, 300 epochs, 2048 samples for LieGG; lower is more equivariant). LEE ranks the provably equivariant model last and an untrained network first; LieGG and CKALE recover the correct order. The $\|\nabla\|$ column is the mechanism: LEE $= \|L_X f\|/\|f\|$ is invariant to uniformly rescaling the features but not to their being nearly constant, where the numerator vanishes and the denominator does not. The untrained network's input gradient is over $700\times$ smaller than the others'. LieGG normalizes by its polarization matrix's largest singular value, where rescaling cancels, and is unaffected.

	LieGG					
model	sym. var.	null dim	align	$\|\nabla\|$	LEE	$1-$CKA
Equivariant SO(2)	**0.00000**	**1.0**	**1.0000**	5.9228	0.9999	**0.000000**
MLP, augmented	0.05760	0.0	0.0000	5.9187	0.3702	0.008651
MLP, no augmentation	0.06845	0.0	0.0000	6.1034	0.6413	0.042468
MLP, untrained	0.16950	0.0	0.0000	**0.0083**	**0.2518**	0.460116

Table 7: LieGG makes an assumption that the input space has to be equivariant and the output space has to be invariant to rotation. Purposely building a model that has an equivariant output space shows that LieGG is unable to detect it.

		LieGG				
readout	m	sym. var. $\downarrow$	null dim	align $\uparrow$	$A_{\text{true}}/\text{rand} \downarrow$	$1-$CKA $\downarrow$
invariant scalar, `head(||h||)`	1	0.00003	**1.0**	**1.0000**	0.00008	0.000000
invariant vector, `||h||`	64	0.00022	**1.0**	**1.0000**	0.00007	0.000033
equivariant, `h`	128	**0.43667**	**0.0**	**0.0000**	**0.87357**	**0.000000**

SEIS has a similar problem to the Lie derivative method where it is unable to detect the equivariance of steerable networks. We compare SEIS between a network that does spatial re-encoding and the steerable network which does spatial and channel re-encodings. Table 8 shows that SEIS is able to detect equivariance only when it changes the space of the features, but not the channel of the features. SEIS is reported separately because its construction requires an unpooled spatial grid: it cannot be evaluated on the point-cloud experiments, which have no spatial axis, nor on the pooled comparison, where the axis it operates on has been averaged away.

5.5 Robustness of CKALE Across Many Domains

To analyze the performance of CKA on real life models and datasets, we used 6 unique tasks: image recognition, image recolorization, fluid flow prediction on a grid, fluid flow prediction on a point cloud, stress tensor prediction on a grid, and stress tensor prediction on a point cloud. For image recognition, we used CIFAR-10 (Krizhevsky, 2009) and MNIST-Font (Wang et al., 2023) datasets. For colorization we also used CIFAR-10 and STL-10 (Coates et al., 2011). For the fluid flow and stress datasets we generated our own datasets. To generate the fluid flow dataset we generate a bunch of random points with a velocity and temperature in a vertical 2 dimensional slice of space with warm blobs placed lower in the field and cold blobs placed higher in the field. We walk the simulation forward 5 steps and use that as the final result. To generate the stress dataset we randomly generate the stress potential field. To produce symmetry breaking effects we apply a stiffness and a directional shear coupling along the x direction. For any grid-like dataset we apply C_4 transformations during training the augmented models and for any point cloud dataset we apply $SO(2)$ transformations so the model learns $SO(2)$ equivariance.

For the grid-like datasets we use 4 unique models, a convolutional neural network (CNN), a UNet (Ronneberger et al., 2015), a vision transformer (ViT) (Dosovitskiy et al., 2021) and a perceptron. For the point cloud datasets we again use a perceptron, but also a multilayer perceptron, a PointNet (Qi et al., 2017) and a SetTransformer (Lee et al., 2019). None of these models are equivariant which is necessary to see how they may learn equivariance when presented with augmented data.

For each dataset and model, we run 10 seeds. We measure the equivariance error against the test dataset with 90 degree rotations for grid like data and 21.17 degree rotations for pointlike data

Table 8: Comparison on SEIS equivariance measure between spatial and channel re-encoding. SEIS is able to detect spatial re-encodings but not channel re-encodings and is therefore unable to detect equivariance of steerable models.

transformation of the features	$\mathcal{S}_{\text{equiv}}$	$\mathcal{S}_{\text{inv}}$	SEIS reads it as
identity (reference)	1.0000	1.0000	—
spatial re-encoding rotate the grid	**1.0000**	0.1311	*re-encoded* ✓
channel re-encoding the true $\rho(g)$	**0.2238**	0.0941	*destroyed* ×
disjoint images (reference)	0.1361	0.0767	—

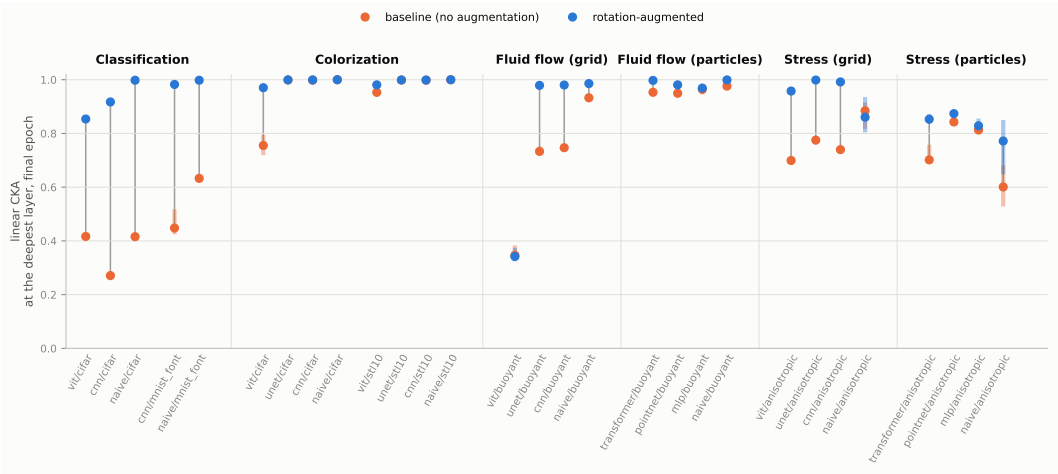

Figure 3: Final-layer CKA across models and datasets for C_4 and $SO(2)$ tasks ("naive" denotes the perceptron). Augmentation increases learned equivariance on most tasks but not colorization, likely because colorization relies more on texture than position. Error bars show the 25th and 75th percentiles.

(16 non-identity angles). Figure 3 shows that for most tasks, augmentation improves last-layer equivariance.

6 Conclusion

We showed that normalized mutual information gives a necessary condition for equivariance that requires no knowledge of the feature-space action, and that it is reliable in small discrete settings. Because MI degrades with large alphabets and is intractable for continuous activations, we introduced CKALE, which is exact under orthogonal feature transformations and scales across tasks and architectures without discovering or assuming the group action.

Limitations and Future Work Mutual information cannot be computed exactly in continuous spaces. CKA is limited to orthogonal transformations, though it is more resilient to random scrambles and covers the transformation class that steerable equivariant networks use. We also subsample infinite groups. Future work could avoid this subsampling and explore kernels that capture non-orthogonal transformations.

AI use statement

In this work, we used generative AI tools for creating code to generate synthetic datasets, to assist in the writing of mathematical claims, critical ingredients for those claims, and the construction of proofs, to design and provide feedback on research methodology, and to implement certain experiments. We have not used generative AI tools to develop theoretical models or conceptual frame-

works, propose or refine hypotheses, or interpret results, and assistance with translation and with cleaning and reformatting datasets is not applicable to this work. Additionally, we used generative AI tools to create scientific figures, suggest experimental parameters, assist with drafting and editing papers, discover related works, and write code. We have reviewed all AI-assisted work. We verified all references from LLMs to ensure they were correct. One author checked the code for correctness and logical results. All authors ensured any information written on the paper was correct. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

Reproducibility statement

All code used to run the experiments in our project was provided in the supplementary materials. Additionally, we supply descriptions of the dataset creation and experiments we ran in Section 5.5.

References

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J. Ballard, Joshua Bambrick, Sebastian W. Bodenstein, David A. Evans, Chia-Chun Hung, Michael O'Neill, David Reiman, Kathryn Tunyasuvunakool, Zachary Wu, Akvilė Žemgulytė, Eirini Arvaniti, Charles Beattie, Ottavia Bertolli, Alex Bridgland, Alexey Cherepanov, Miles Congreve, Alexander I. Cowen-Rivers, Andrew Cowie, Michael Figurnov, Fabian B. Fuchs, Hannah Gladman, Rishub Jain, Yousuf A. Khan, Caroline M. R. Low, Kuba Perlin, Anna Potapenko, Pascal Savy, Sukhdeep Singh, Adrian Stecula, Ashok Thillaisundaram, Catherine Tong, Sergei Yakneen, Ellen D. Zhong, Michal Zielinski, Augustin Žídek, Victor Bapst, Pushmeet Kohli, Max Jaderberg, Demis Hassabis, and John M. Jumper. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016): 493–500, 2024. doi: 10.1038/s41586-024-07487-w. URL https://doi.org/10.1038/s41586-024-07487-w.

Diane Bouchacourt, Mark Ibrahim, and Ari S. Morcos. Grounding inductive biases in natural images: invariance stems from variations in data. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/a2fe8c05877ec786290dd1450c3385cd-Abstract.html.

Robert-Jan Bruintjes, Tomasz Motyka, and Jan van Gemert. What affects learned equivariance in deep image recognition models? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 4839–4847, 2023. doi: 10.1109/CVPRW59228.2023.00512. URL https://arxiv.org/abs/2304.02628.

Chai Discovery. Chai-1: Decoding the molecular interactions of life. *bioRxiv*, 2024. doi: 10.1101/2024.10.10.615955. URL https://www.biorxiv.org/content/early/2024/10/11/2024.10.10.615955.

Hippolyte Charvin, Nicola Catenacci Volpi, and Daniel Polani. Towards information theory-based discovery of equivariances. In *NeurIPS 2023 Workshop on Symmetry and Geometry in Neural Representations*, 2023. URL https://openreview.net/forum?id=oD8DD5jQ5I.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 215–223. PMLR, 11–13 Apr 2011. URL https://proceedings.mlr.press/v15/coates11a.html.

MohammadReza Davari, Stefan Horoi, Amine Natik, Guillaume Lajoie, Guy Wolf, and Eugene Belilovsky. Reliability of CKA as a similarity measure in deep learning, 2022. URL https://arxiv.org/abs/2210.16156.

Frances Ding, Jean-Stanislas Denain, and Jacob Steinhardt. Grounding representation similarity through statistical testing. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/0c0bf917c7942b5a08df71f9da626f97-Abstract.html.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Arthur Gretton, Kenji Fukumizu, Choon Teo, Le Song, Bernhard Schölkopf, and Alex Smola. A kernel statistical test of independence. In *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/file/d5cfead94f5350c12c322b5b664544c1-Paper.pdf.

Nate Gruver, Marc Anton Finzi, Micah Goldblum, and Andrew Gordon Wilson. The Lie derivative for measuring learned equivariance. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=JL7Va5Vy15J.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3519–3529. PMLR, 2019. URL https://proceedings.mlr.press/v97/kornblith19a.html.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL https://cave.cs.toronto.edu/kriz/learning-features-2009-TR.pdf.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam R. Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3744–3753. PMLR, 2019. URL https://proceedings.mlr.press/v97/lee19d.html.

Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 991–999, 2015. URL https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Lenc_Understanding_Image_Representations_2015_CVPR_paper.html.

Huahua Lin, Katayoun Farrahi, and Xiaohao Cai. SEIS: Subspace-based equivariance and invariance scores for neural representations, 2026. URL https://arxiv.org/abs/2602.04054.

Marco Möller and Daniel Polani. Emergence of common concepts, symmetries and conformity in agent groups—an information-theoretic model. *Interface Focus*, 13(3):20230006, 2023. doi: 10.1098/rsfs.2023.0006. URL https://doi.org/10.1098/rsfs.2023.0006.

Artem Moskalev, Anna Sepliarskaia, Ivan Sosnovik, and Arnold W. M. Smeulders. LieGG: Studying learned Lie group generators. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://openreview.net/forum?id=9sKZ60VtRmi.

Artem Moskalev, Anna Sepliarskaia, Erik J Bekkers, and Arnold W.M. Smeulders. On genuine invariance learning without weight-tying. In *Proceedings of 2nd Annual Workshop on Topology, Algebra, and Geometry in Machine Learning (TAG-ML)*, volume 221 of *Proceedings of Machine Learning Research*, pp. 218–227. PMLR, 28 Jul 2023. URL https://proceedings.mlr.press/v221/moskalev23a.html.

Alex Murphy, Joel Zylberberg, and Alona Fyshe. Correcting biased centered kernel alignment measures in biological and artificial neural networks, 2024. URL https://arxiv.org/abs/2405.01012. ICLR 2024 Re-Align Workshop.

Thao Nguyen, Maithra Raghu, and Simon Kornblith. Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=KJNcAkY8tY4.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 652–660, 2017. URL https://arxiv.org/abs/1612.00593.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, volume 30, pp. 6076–6085, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/dc6a7e655d7e5840e66733e9ee67cc69-Abstract.html.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351, pp. 234–241. Springer, Cham, 2015. doi: 10.1007/978-3-319-24574-4_28. URL https://doi.org/10.1007/978-3-319-24574-4_28.

Claude Elwood Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948. doi: 10.1002/j.1538-7305.1948.tb01338.x. URL https://doi.org/10.1002/j.1538-7305.1948.tb01338.x.

Le Song, Alex Smola, Arthur Gretton, Justin Bedo, and Karsten Borgwardt. Feature selection via dependence maximization. *Journal of Machine Learning Research*, 13(47):1393–1434, 2012. URL https://jmlr.org/papers/v13/song12a.html.

Dian Wang, Xupeng Zhu, Jung Yeon Park, Mingxi Jia, Guanang Su, Robert Platt, and Robin Walters. A general theory of correct, incorrect, and extrinsic equivariance. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=2FMJtNDLeE.

Maurice Weiler and Gabriele Cesa. General e(2)-equivariant steerable cnns. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/45d6637b718d0f24a237069fe41b0db4-Paper.pdf.

Alex H. Williams, Erin Kunz, Simon Kornblith, and Scott W. Linderman. Generalized shape metrics on neural representations. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/252a3dbaeb32e7690242ad3b556e626b-Abstract.html.

Jeremy Wohlwend, Gabriele Corso, Saro Passaro, Noah Getz, Mateo Reveiz, Ken Leidal, Wojtek Swiderski, Liam Atkinson, Tally Portnoi, Itamar Chinn, Jacob Silterra, Tommi Jaakkola, and Regina Barzilay. Boltz-1: Democratizing biomolecular interaction modeling. *bioRxiv*, 2024. doi: 10.1101/2024.11.19.624167. URL https://doi.org/10.1101/2024.11.19.624167.

Jianke Yang, Robin Walters, Nima Dehmamy, and Rose Yu. Generative adversarial symmetry discovery. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 39488–39508. PMLR, 2023. URL https://proceedings.mlr.press/v202/yang23n.html.

A LEMMAS

Lemma 1. $\mathrm{NMI}(X,Y) = 1 \iff H(X|Y) = H(Y|X) = 0.$

Proof. By definition of mutual information, $I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$, therefore $2I(X;Y) = H(X) + H(Y) - (H(X|Y) + H(Y|X))$.

If $\mathrm{NMI}(X,Y) = 1$, then by definition $\frac{2I(X;Y)}{H(X)+H(Y)} = \frac{H(X)+H(Y)-(H(X|Y)+H(Y|X))}{H(X)+H(Y)} = 1 - \frac{H(X|Y)+H(Y|X)}{H(X)+H(Y)} = 1$, so $\frac{H(X|Y)+H(Y|X)}{H(X)+H(Y)} = 0$ which requires $H(X|Y) + H(Y|X) = 0$

If $H(X|Y) = H(Y|X) = 0$, then $2I(X;Y) = H(X) + H(Y) - (H(X|Y) + H(Y|X)) = H(X) + H(Y)$. Therefore $\mathrm{NMI}(X, Y) = 1$. □

Lemma 2. *Under the nondegeneracy assumption $\tilde{X} \neq 0$ and $\tilde{Y} \neq 0$,*

$$\mathrm{CKA}(K_X, K_Y) = 1 \iff \tilde{X} = c\,\tilde{Y} \text{ for some } c > 0,$$

where $\tilde{X} = HK_XH$ and $\tilde{Y} = HK_YH$ are the centered Gram matrices.

Proof. Vectorize the centered Gram matrices and treat them as elements of $\mathbb{R}^{n\times n}$ with the Frobenius inner product $\langle A, B\rangle_F = \mathrm{tr}(A^\top B)$. By definition, CKA is the cosine of the angle between them: $\mathrm{CKA}(K_{\mathbf{X}}, K_{\mathbf{Y}}) = \dfrac{\langle \tilde{X}, \tilde{Y}\rangle_F}{\|\tilde{X}\|_F \, \|\tilde{Y}\|_F}$.

By the Cauchy–Schwarz inequality, $|\langle \tilde{X}, \tilde{Y}\rangle_F| \leq \|\tilde{X}\|_F \, \|\tilde{Y}\|_F$, so $\mathrm{CKA} \in [-1, 1]$, with equality $|\mathrm{CKA}| = 1$ iff $\tilde{X}$ and $\tilde{Y}$ are linearly dependent, i.e. $\tilde{X} = c\,\tilde{Y}$ for some $c \neq 0$ (nonzero by nondegeneracy).

Since $K_{\mathbf{X}}, K_{\mathbf{Y}}$ are Gram matrices they are positive semidefinite, and centering preserves this, so $\langle \tilde{X}, \tilde{Y}\rangle_F = \mathrm{tr}(\tilde{X}\tilde{Y}) = \mathrm{tr}\big(\tilde{X}^{1/2}\tilde{Y}\,\tilde{X}^{1/2}\big) \geq 0$. Hence the numerator is nonnegative and $\mathrm{CKA} \in [0, 1]$.

Combining the two: $\mathrm{CKA} = 1$ forces the Cauchy–Schwarz equality $\tilde{X} = c\,\tilde{Y}$ together with $\langle \tilde{X}, \tilde{Y}\rangle_F \geq 0$. Substituting gives $\langle \tilde{X}, \tilde{Y}\rangle_F = c\,\|\tilde{Y}\|_F^2$, which is nonnegative only when $c > 0$. Conversely, if $\tilde{X} = c\,\tilde{Y}$ with $c > 0$, direct substitution yields $\mathrm{CKA} = 1$. □

B CKA AND ALIASING

B.1 A RESAMPLING FLOOR FOR THE CONTINUOUS SWEEP

The sweep of Figure 1 compares $f(x)$ against $f(\mathrm{rot}(\theta)x)$ at arbitrary θ, and a non-lattice rotation of a pixel grid is not a permutation. It requires interpolation, which low-pass filters the image and discards content at the corners, so $\mathrm{rot}(\theta)x$ is not "x, rotated" but a rotated and slightly damaged x. The rising error away from the group elements could therefore reflect the probe damaging its own input rather than the model failing to be equivariant.

That possibility can be measured directly. Linear CKA is invariant to orthogonal maps and a quarter turn is a permutation of pixels, so $1 - \mathrm{CKA}(x, \mathrm{rot}(\theta)x)$ evaluated on *raw pixels with no network* is exactly zero on the lattice and positive off it. It isolates the rotation operator's departure from being an isometry, with the model removed.

Table 9 reports it. The floor behaves as the argument requires: exactly zero at the four quarter turns, 0.00011–0.00013 elsewhere. The model's error sits about $3\times$ above it, so the convolution amplifies the input damage rather than merely inheriting it. But both quantities are of order 10^{-4}, and both fall to zero by a 113^2 grid, so the entire off-lattice signal at these angles is a sampling artifact rather than a property of the model.

The peaks in the continuous sweep run from 0.04 for C16 to 0.51 for C1, between two and three orders of magnitude above this floor, so the group structure recovered there is genuine signal. We report the negative because the confound was plausible and the floor had not previously been measured at the input, only at the model.

B.2 CKA DOES NOT DETECT ALIASING

The same experiment read the other way exposes a limitation. Interpolation genuinely destroys information—it is irreversible, and the representation-aware equivariance error at these angles is two to three orders of magnitude larger than what CKA reports. Downsampling the feature map at export makes the point without any rotation at all.

Naive subsampling inflates the equivariance error by 62%, from 0.13429 to 0.21735, and prefiltering before subsampling brings it back to 0.15250—a loss identified by mechanism and then repaired.

Table 9: The resampling floor. **(a)** A C16 steerable conv at its own sixteen group elements: the input floor is $1 - \mathrm{CKA}$ between raw pixels and their own rotation, the model column the same quantity on the conv's features. **(b)** Both, by grid resolution. All entries are $1 - \mathrm{CKA}$; 128 MNIST images, 29^2 grid in panel (a).

(a) at the C16 model's own group elements

rotation	on lattice?	input floor	model	model/floor
$0°, 90°, 180°, 270°$	yes	**0.00000**	**0.00000**	—
$22.5°$ and its 3 repeats	no	0.00011	0.00032	2.83
$67.5°$ and its 3 repeats	no	0.00011	0.00026	2.30
$45°$ and its 3 repeats	no	0.00013	0.00054	4.10
off-lattice mean		0.00012	0.00037	3.08

(b) the same quantities, by grid resolution

	$22.5°$		$45.0°$		$90°$
grid	floor	model	floor	model	(both)
29^2	0.00011	0.00032	0.00013	0.00054	0.00000
57^2	0.00000	0.00004	0.00000	0.00005	0.00000
113^2	0.00000	0.00000	0.00000	0.00000	0.00000
225^2	0.00000	0.00000	0.00000	0.00000	0.00000

Table 10: Downsampling a feature map at export aliases it. Equivariance error is $\|f(gx) - \rho(g)f(x)\|/\|\rho(g)f(x)\|$ with $\rho(g)$ read off escnn's declared `FieldType`, so it is ground truth rather than a competing estimate. The null is the shuffled control: the pairing between the clean and rotated branches is destroyed while marginals, dimensionality and sample count are held fixed. $\theta = 2.0$ rad ($114.6°$), 256 MNIST images.

condition	equivariance error ↓	linear CKA ↑	CKA null
grid-aligned reference ($\theta = \pi/2$)	**0.00000**	**1.0000**	0.0729
full resolution	0.13429	0.9994	0.0722
subsample	**0.21735**	0.9995	0.0727
blur, then subsample	0.15250	0.9989	0.0432

Linear CKA reads ~ 0.999 through all three and registers neither the damage nor the fix. The baseline is not zero because $\theta = 2.0$ rad is off-lattice and a C_N model on a square grid is exact only at quarter turns, so 0.13429 is the interpolation floor above which the 62% is measured.

The null makes the failure precise. A reading of 0.999 could mean either that the two feature sets really are nearly identical or that the metric returns 0.999 on anything—the second being the degenerate regime RBF CKA enters at small bandwidth, where signal and shuffled control both reach 1.0000. Row-shuffling gives ~ 0.07 here, so CKA sits roughly $14\times$ above its own floor and the readings are real. The failure is one of *range* rather than of validity: across the whole experiment CKA spans 1.0000 to 0.9980 while the quantity it tracks runs from exactly zero to 0.236.

It is not, then, that lost data degrades CKA's resolution. It is that CKA barely notices data has been lost at all, because interpolation perturbs every feature slightly while leaving their pairwise similarities almost intact.

C Progression of Models' Learned Equivariance

Using the same setup as described in Section 5.5, we showcase each layer of every model across all epochs to show how learned equivariance evolves over time.

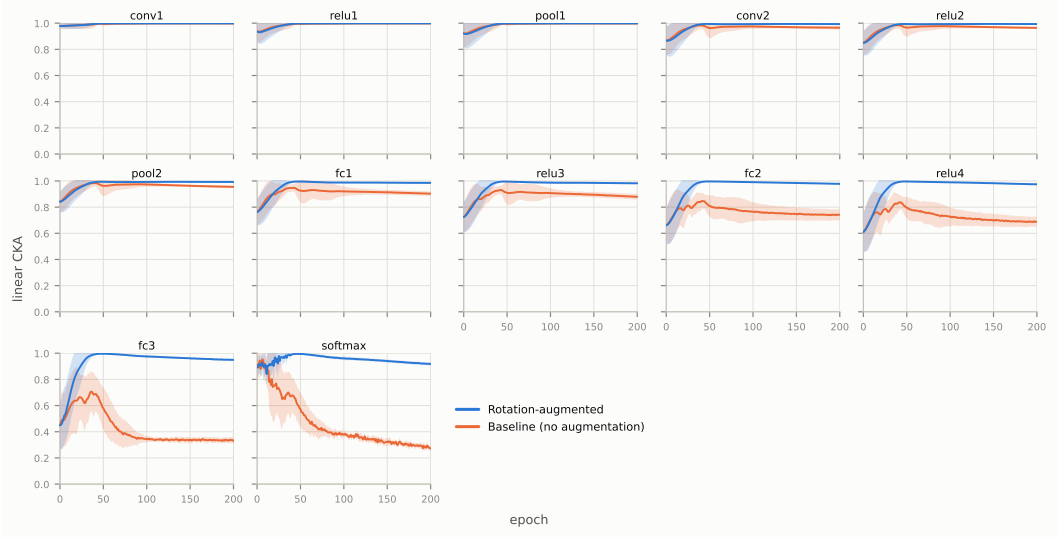

Figure 4: Full view of a CNN progression on learning equivariance on CIFAR-10 classification.

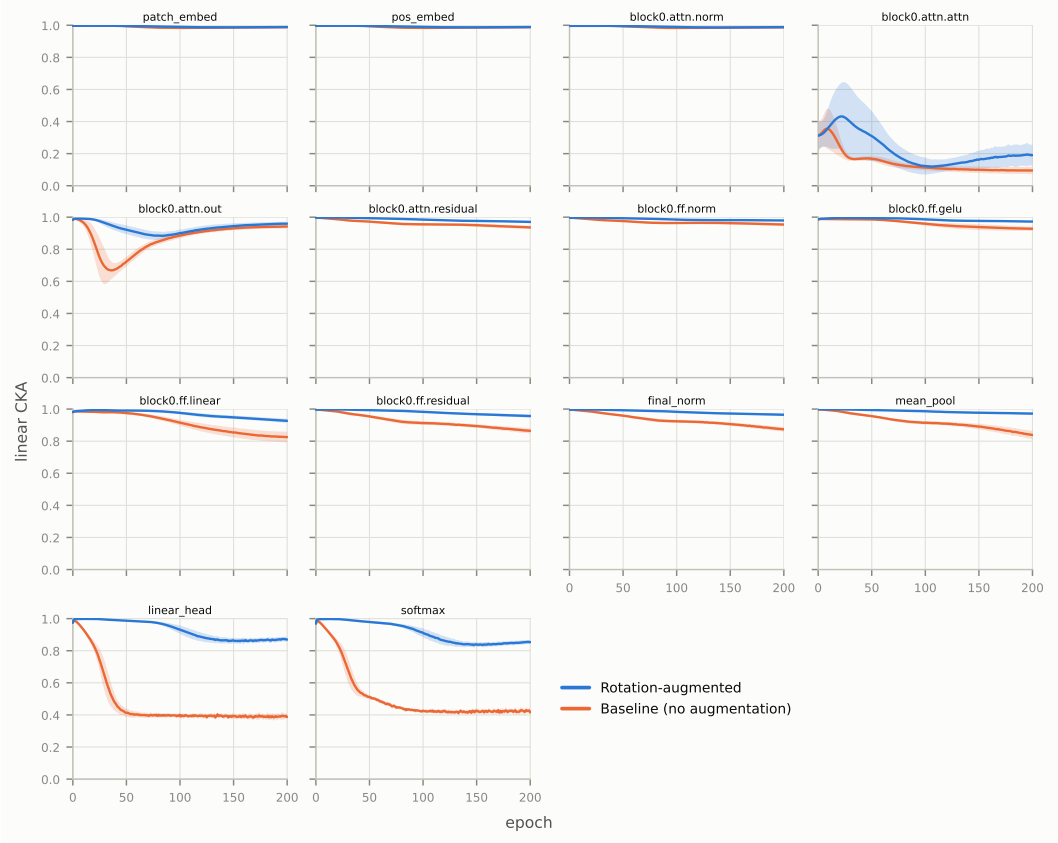

Figure 5: Full view of a ViT progression on learning equivariance on CIFAR-10 classification.

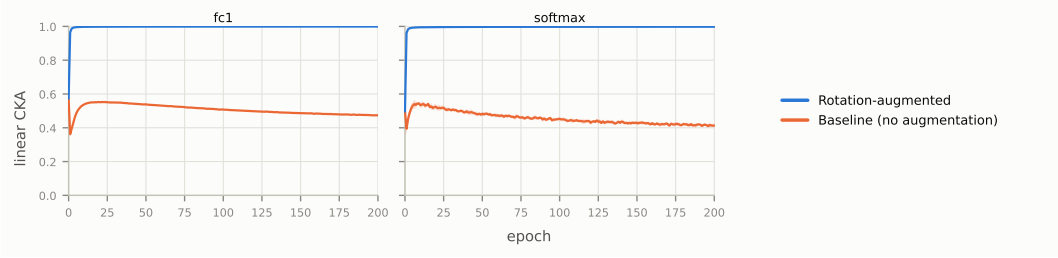

Figure 6: Full view of a perceptron progression on learning equivariance on CIFAR-10 classification.

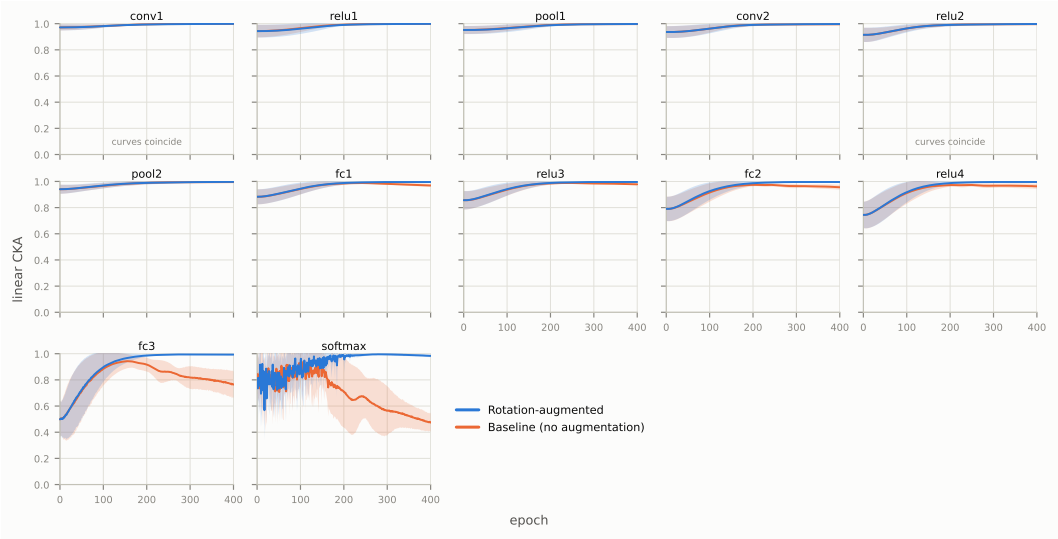

Figure 7: Full view of a CNN progression on learning equivariance on MNIST-Font classification.

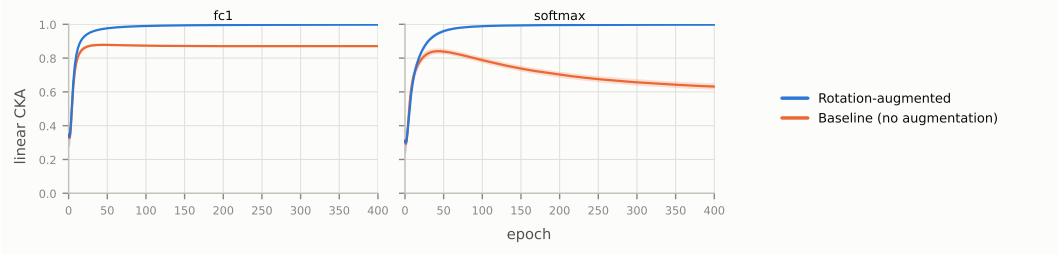

Figure 8: Full view of a perceptron progression on learning equivariance on MNIST-Font classification.

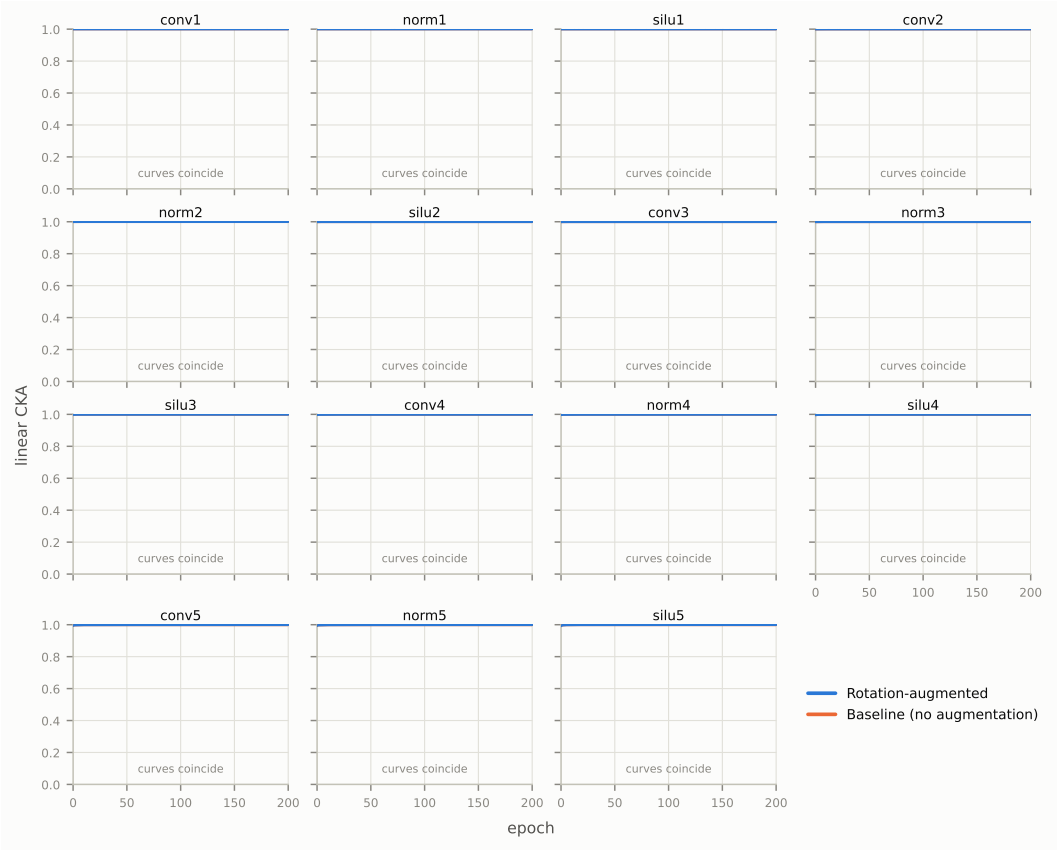

Figure 9: Full view of a CNN learning equivariance on CIFAR-10 colorization.



Figure 10: Full view of a UNet learning equivariance on CIFAR-10 colorization.

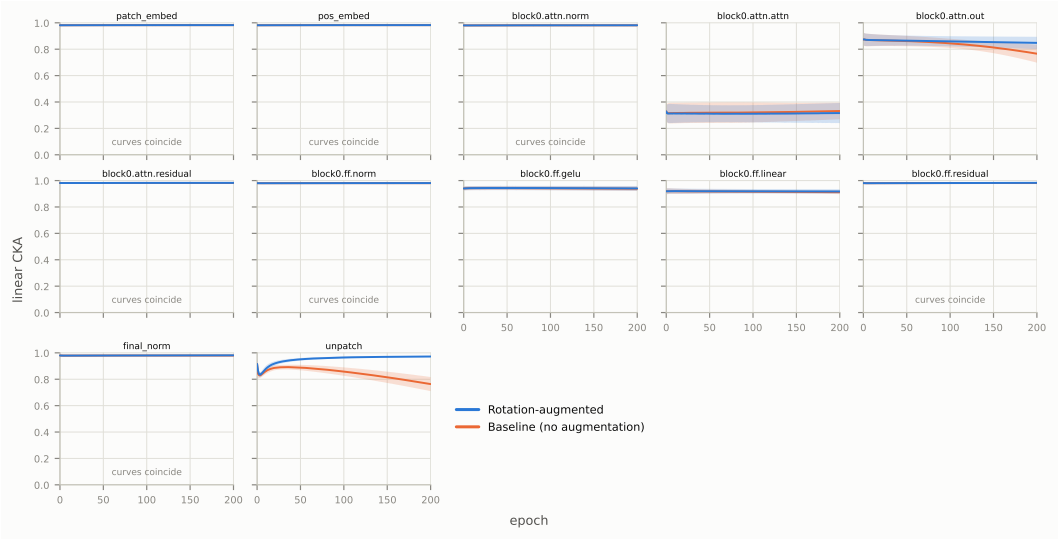

Figure 11: Full view of a ViT learning equivariance on CIFAR-10 colorization.

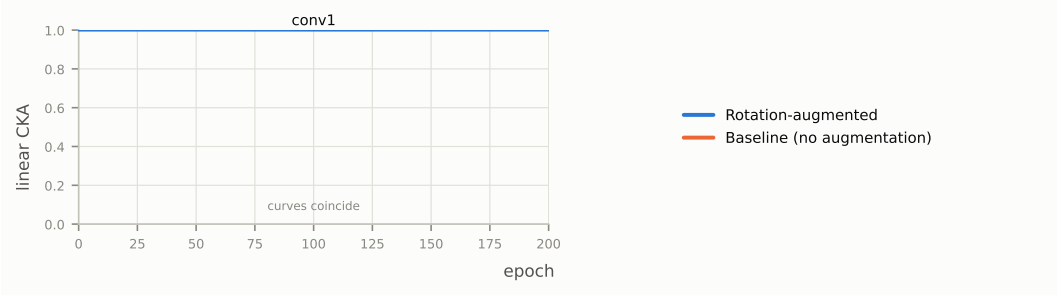

Figure 12: Full view of a perceptron learning equivariance on CIFAR-10 colorization.

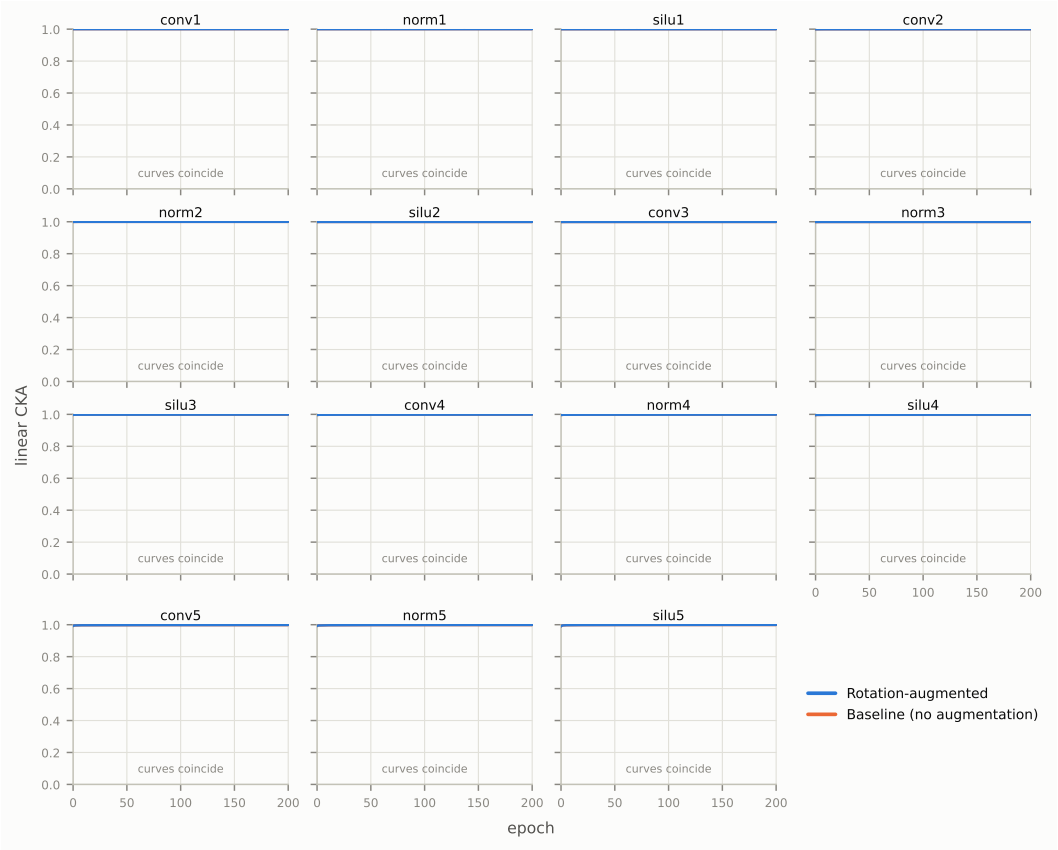

Figure 13: Full view of a CNN learning equivariance on STL-10 colorization.

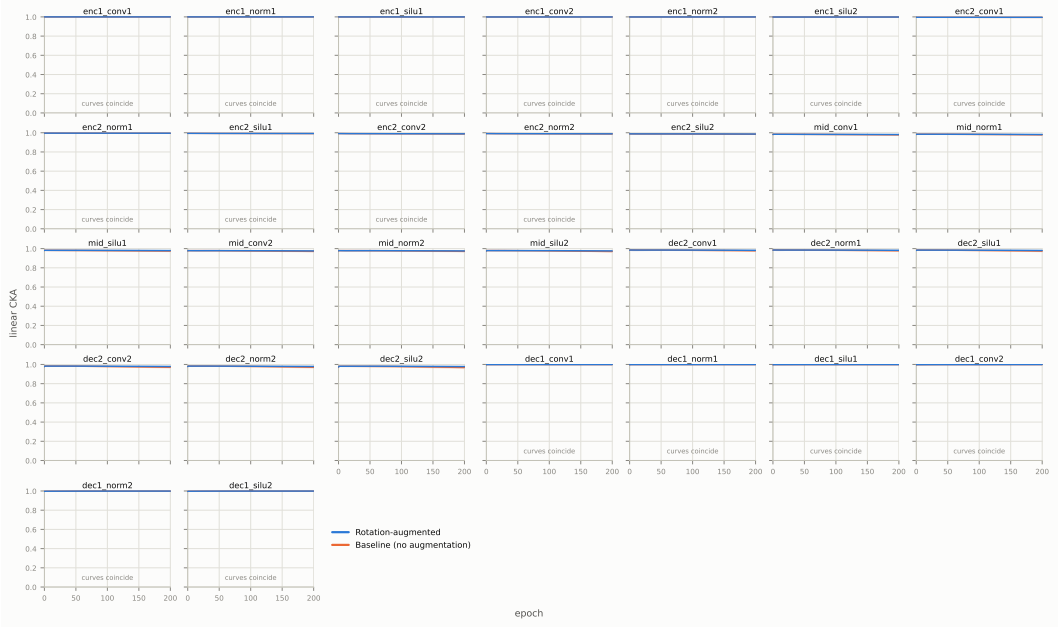

Figure 14: Full view of a UNet learning equivariance on STL-10 colorization.

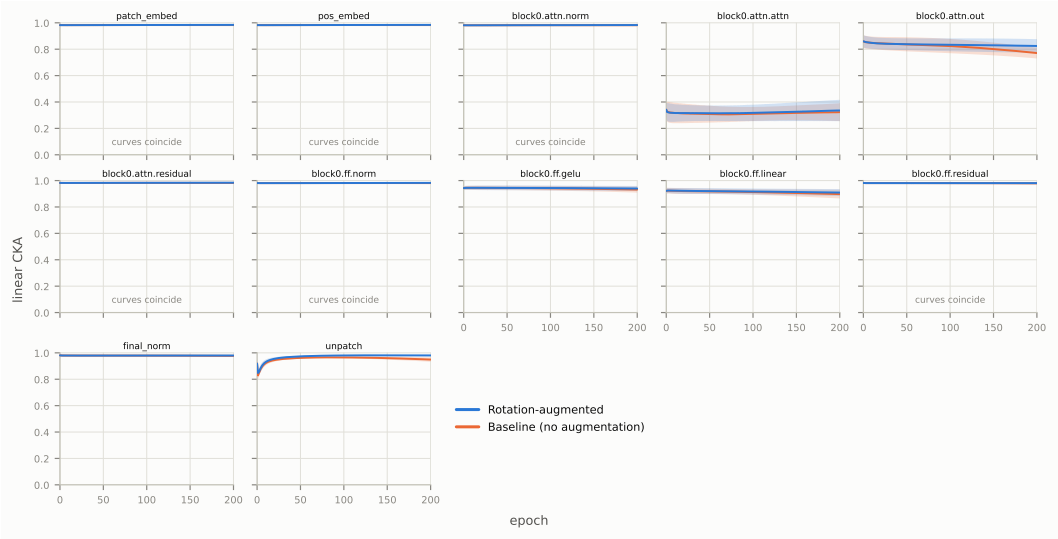

Figure 15: Full view of a ViT learning equivariance on STL-10 colorization.

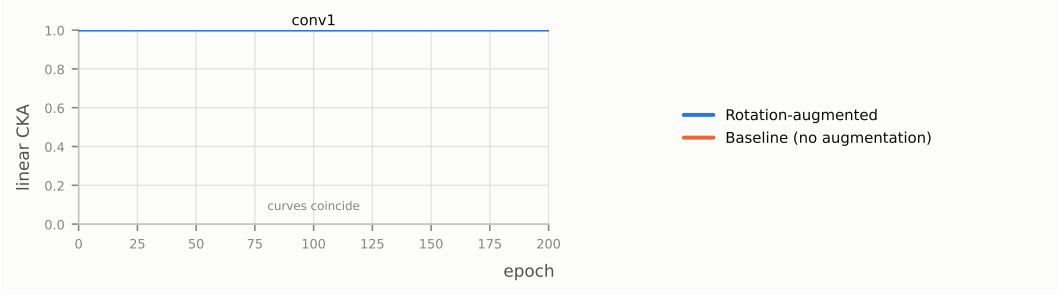

Figure 16: Full view of a perceptron learning equivariance on STL-10 colorization.

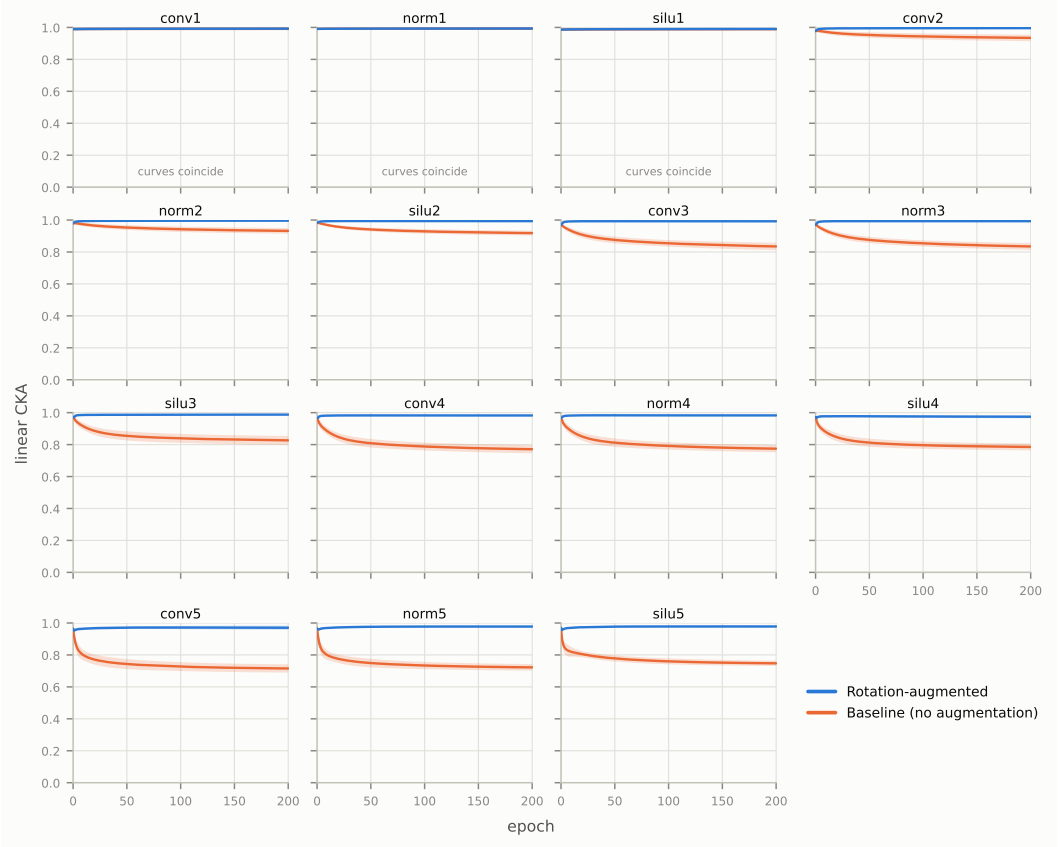

Figure 17: Full view of a CNN learning fluid dynamics on a grid-like buoyancy dataset.

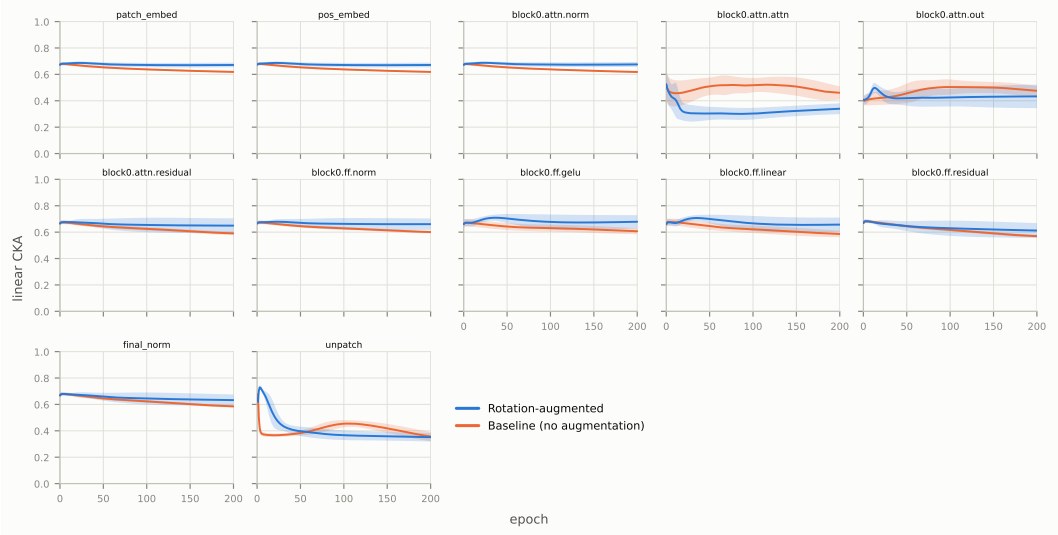

Figure 18: Full view of a ViT learning fluid dynamics on a grid-like buoyancy dataset.

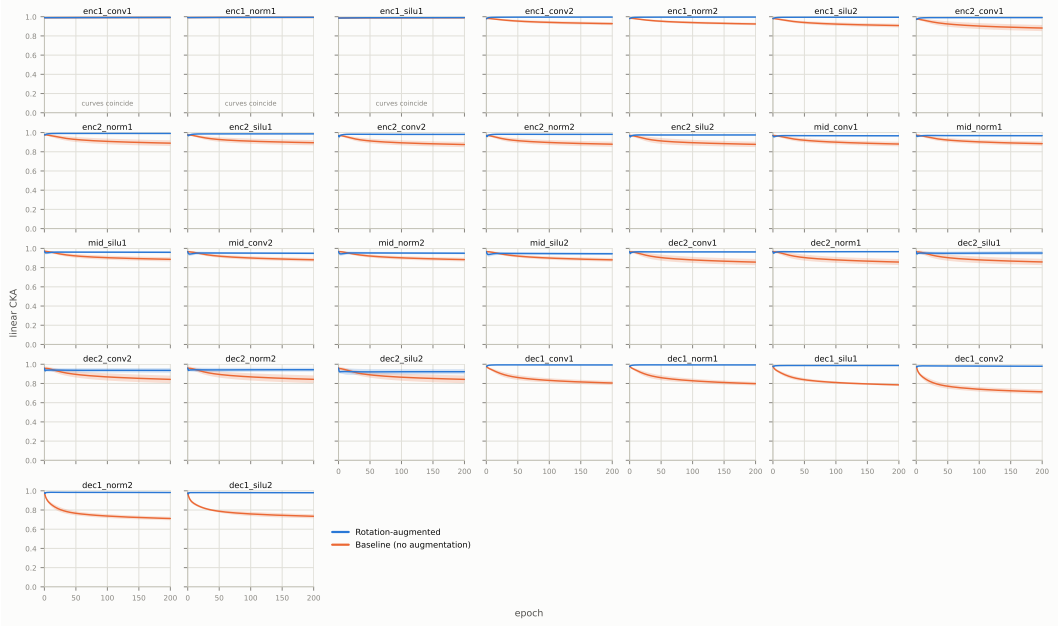

Figure 19: Full view of a UNet learning fluid dynamics on a grid-like buoyancy dataset.

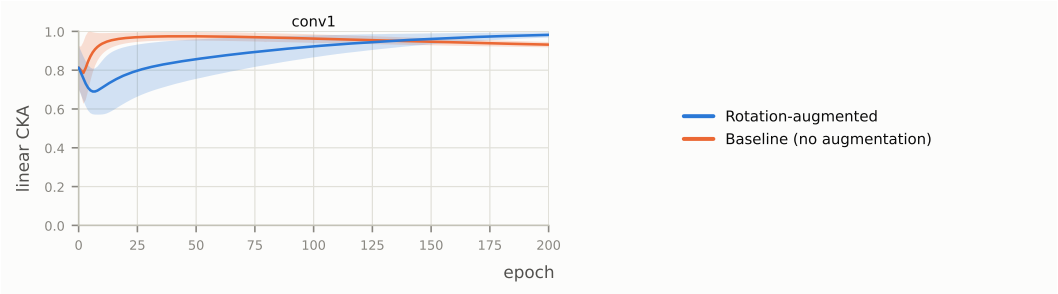

Figure 20: Full view of a perceptron learning fluid dynamics on a grid-like buoyancy dataset.

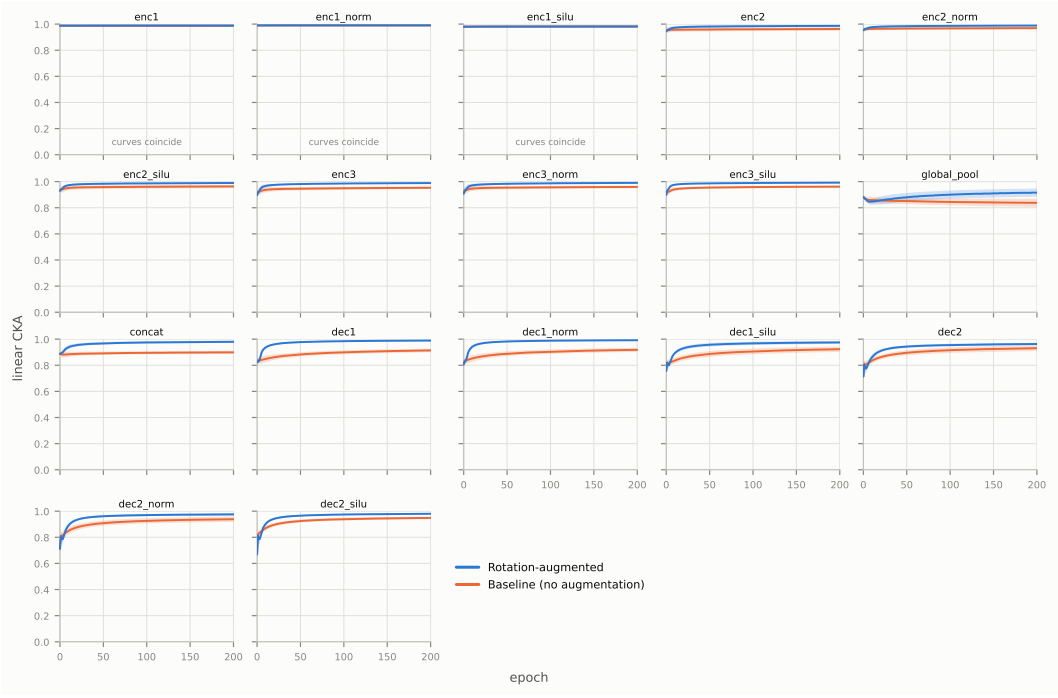

Figure 21: Full view of a PointNet learning fluid dynamics on a particle-like buoyancy dataset.

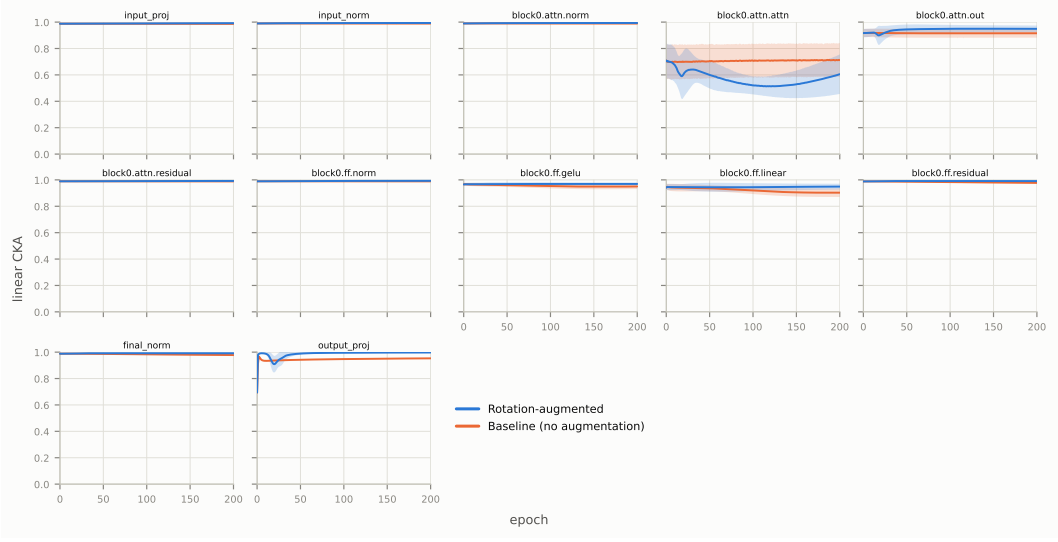

Figure 22: Full view of a SetTransformer learning fluid dynamics on a particle-like buoyancy dataset.

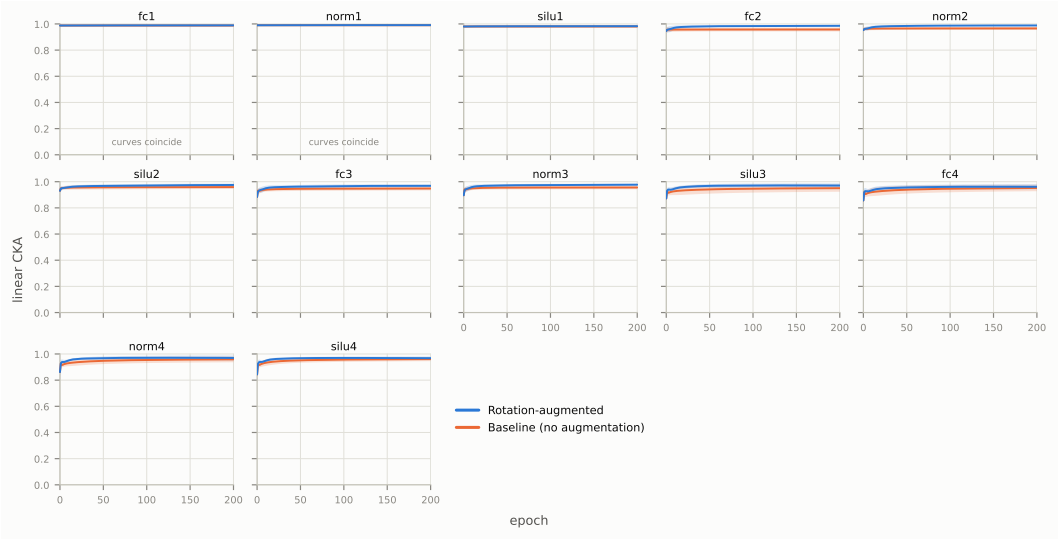

Figure 23: Full view of an MLP learning fluid dynamics on a particle-like buoyancy dataset.

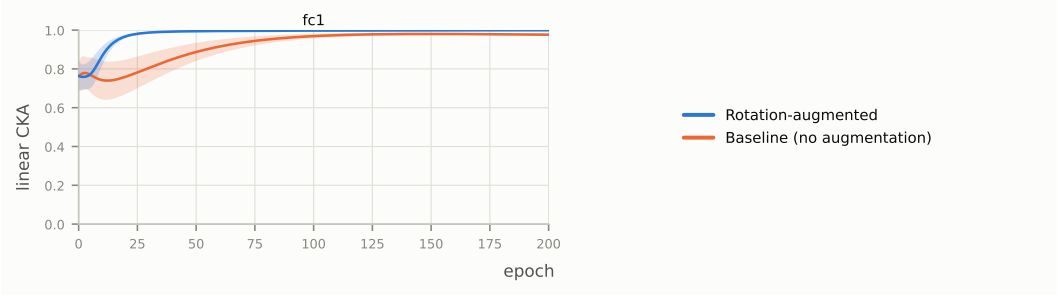

Figure 24: Full view of a perceptron learning fluid dynamics on a particle-like buoyancy dataset.

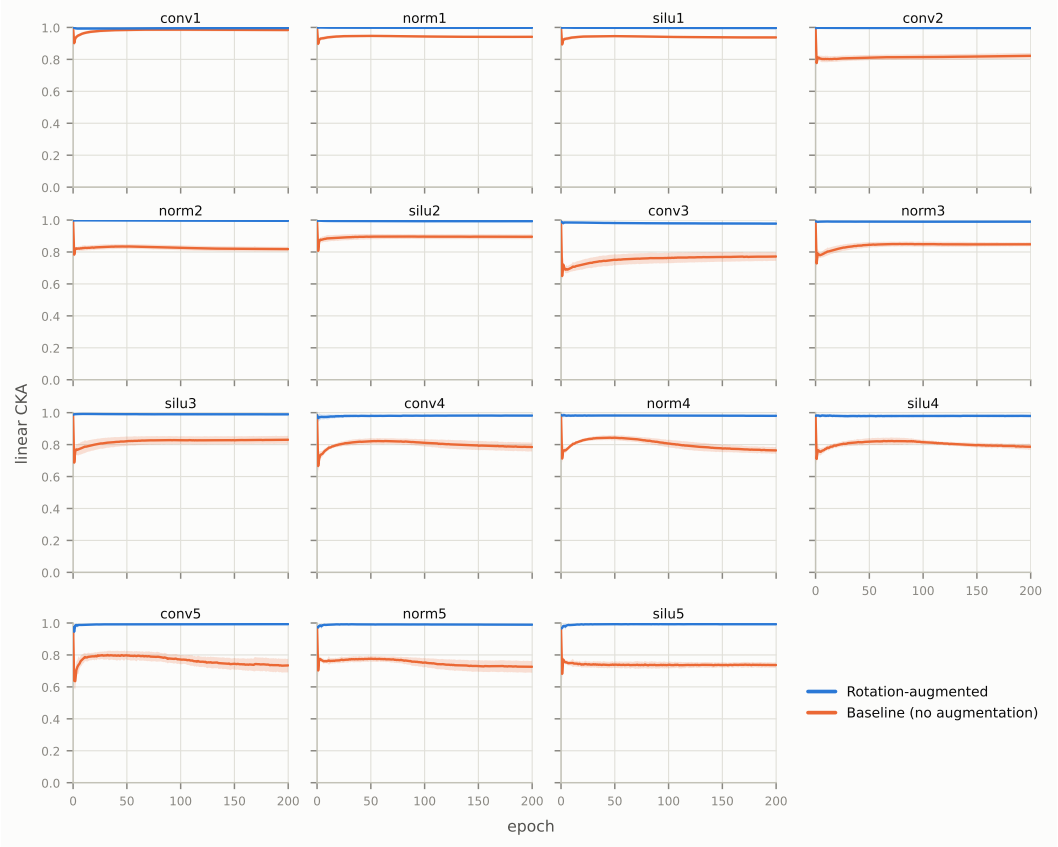

Figure 25: Full view of a CNN learning stress tensor prediction on a grid-like anisotropic dataset.

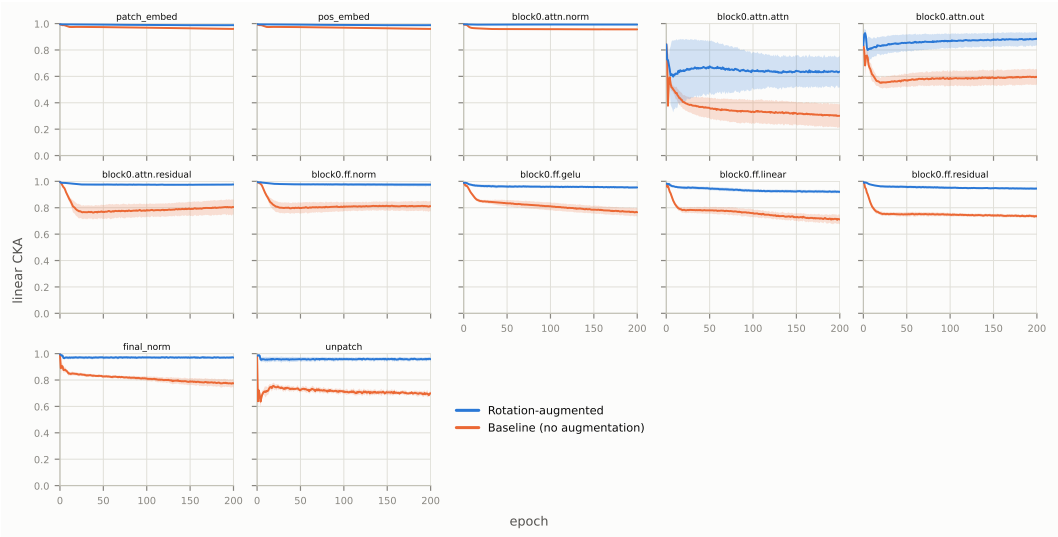

Figure 26: Full view of a ViT learning stress tensor prediction on a grid-like anisotropic dataset.

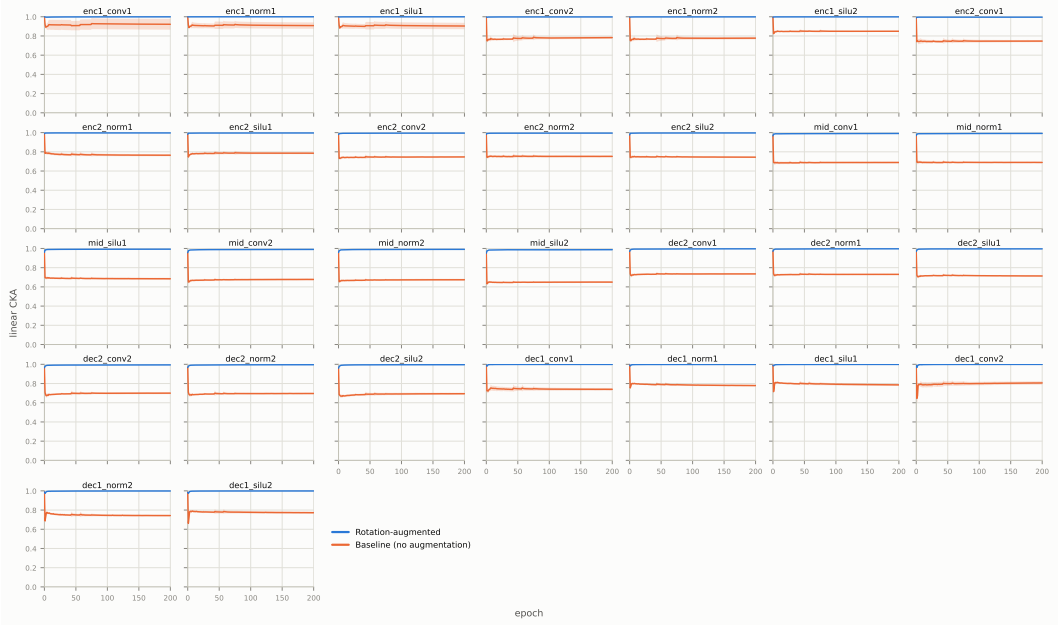

Figure 27: Full view of a UNet learning stress tensor prediction on a grid-like anisotropic dataset.

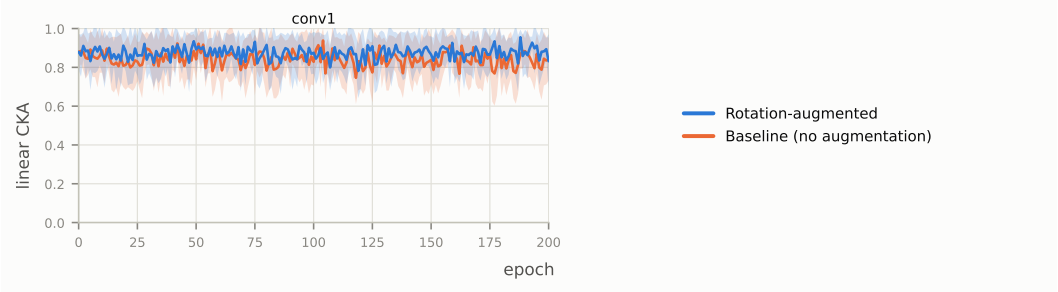

Figure 28: Full view of a perceptron learning stress tensor prediction on a grid-like anisotropic dataset.

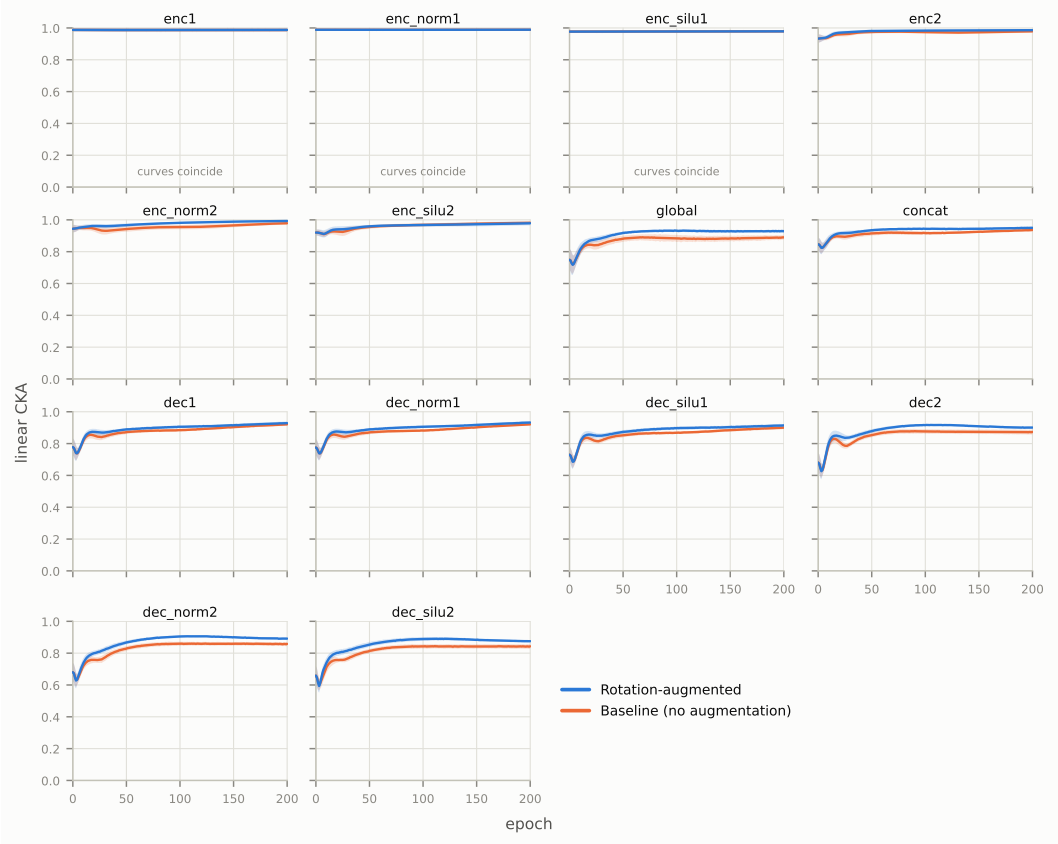

Figure 29: Full view of a PointNet learning stress tensor prediction on a particle-like anisotropic dataset.

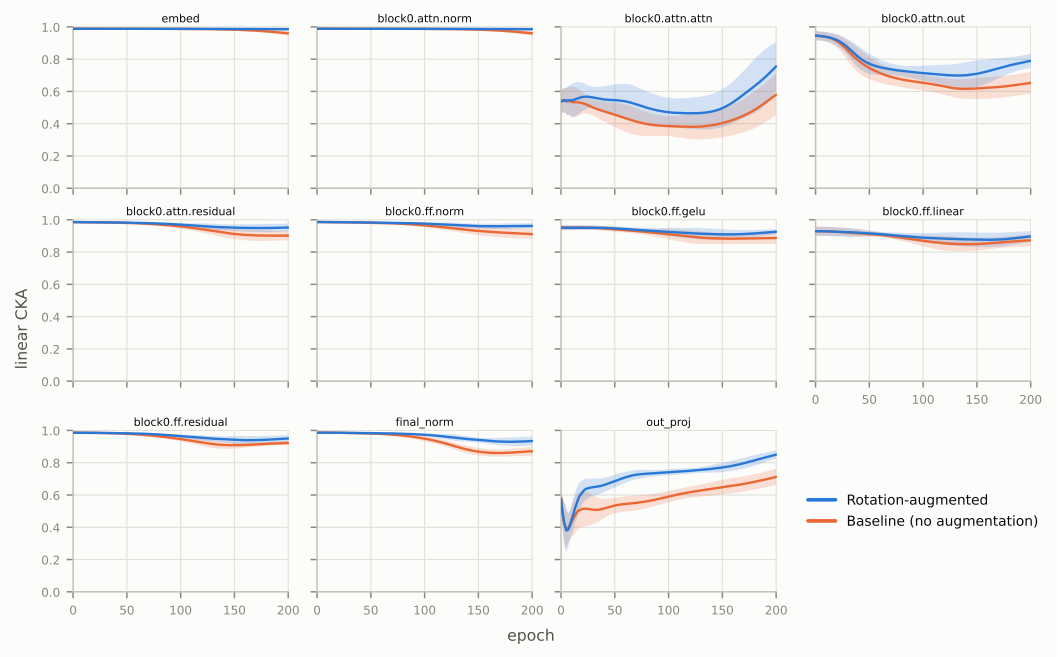

Figure 30: Full view of a SetTransformer learning stress tensor prediction on a particle-like anisotropic dataset.

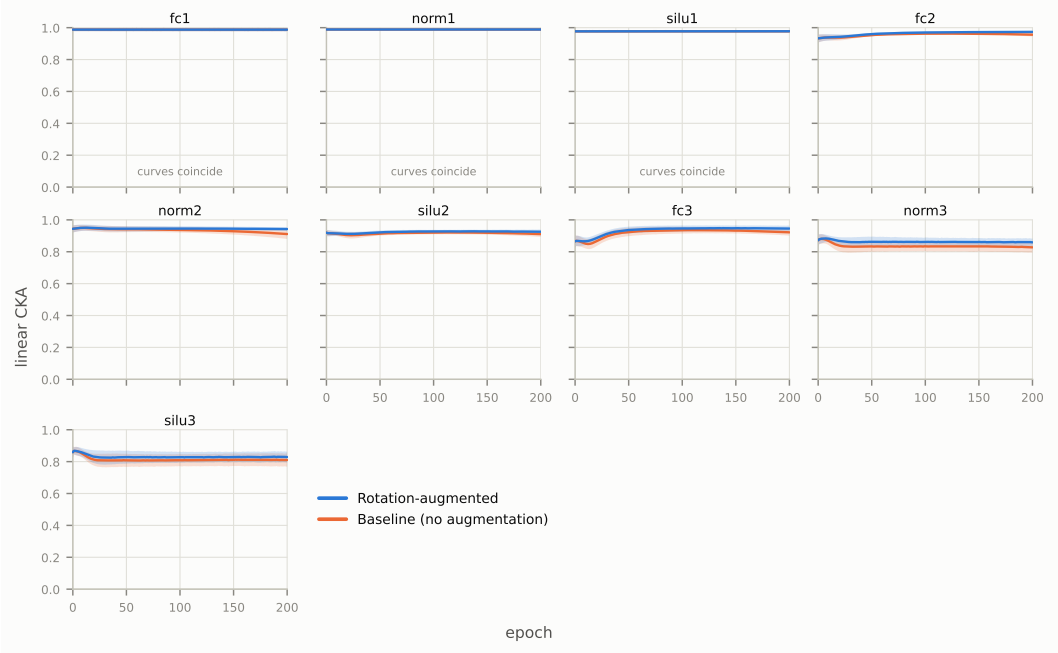

Figure 31: Full view of an MLP learning stress tensor prediction on a particle-like anisotropic dataset.

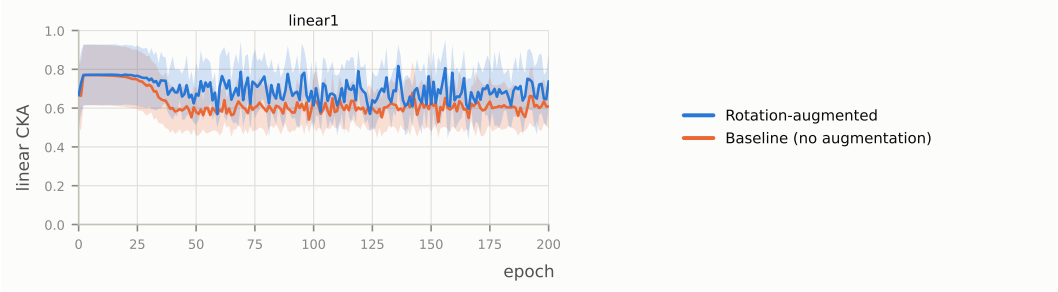

Figure 32: Full view of a perceptron learning stress tensor prediction on a particle-like anisotropic dataset.