Generalization vs. Memorization in Partial Differential Equation Emulators: Or, Training Dynamics of Cross-Time and Cross-Class Gradient Alignment

James Amarel [1] **Nicolas Hengartner** [1] **Robyn Miller** [1] **Kamaljeet Singh** [2] **Siddharth Mansingh** [1]
Arvind Mohan [1] **Benjamin Migliori** [1] **Emily Casleton** [1] **Alexei Skurikhin** [1] **Earl Lawrence** [1] **Gerd J. Kunde** [1]

Abstract

Neural emulators of solution operators to partial differential equations capable of reliable generalization stand to accelerate scientific discovery by extrapolating beyond training regimes and performing over varied initial condition classes. Assessing model generalization capabilities requires distinguishing solution operator-consistent behavior from memorization of feature derived correlations. This can be achieved by using influence functions to characterize how information is shared across examples with different temporal indices and/or of different initial conditions. We compute the hat-matrix metric-weighted gradient alignment by inverting the relevant Hessian restricted to the mini-batch gradient-spanned subspace, enabling exact measurement of cross-time and cross-class coherence during training. This diagnostic is architecture-agnostic and can be efficiently computed. Our results indicate that gradient alignment between training examples is governed by their feature-space similarity. We find that gradient coherence across initial condition classes is suppressed due to feature-space dissimilarity. Moreover, we quantify the temporal decay of gradient alignment induced by standard one-step autoregressive training, revealing that gradient coherence is supressed across time, which limits long-horizon extrapolation performance. Cross-time influence exhibits a negative correlation with rollout error across rollout horizon, indicating that the measured response geometry contains information predictive of inference-time degradation.

[1]Los Alamos National Laboratory, Los Alamos, NM 87545
[2]The University of Arizona, Tucson, AZ, 85721. Correspondence to: James Amarel <jlamarel@lanl.gov>.

Proceedings of the AI4Physics Workshop at the 43rd International Conference on Machine Learning (AI4Physics@ICML 2026), Seoul, South Korea. 2026. Copyright 2026 by the author(s).

1. Introduction

The machine learning of surrogate models has emerged as a powerful technique for efficiently approximating the solutions of computationally intensive partial differential equations (PDEs). These emulation methods range from purely data-driven approaches, trained on high-fidelity simulation data, to physics-informed neural networks, which integrate PDE structures directly into the training loss, enforcing physical laws as soft constraints (Raissi et al., 2019). Such models hold promise for achieving significant computational acceleration in applications such as fluid dynamics (Takamoto et al., 2024; Lippe et al., 2023; Gupta & Brandstetter, 2022; Ohana et al., 2025; Herde et al., 2024), climate modeling (Bodnar et al., 2024), and materials science (Batatia et al., 2024), enabling rapid research and development across diverse scientific and engineering disciplines. Despite these advances, reliable generalization and robustness remains a critical challenge (Krishnapriyan et al., 2021). Distinguishing between memorization of training examples and generalization is critical to evaluating model robustness; diagnostic tools such as influence functions (Koh & Liang, 2020; Bae et al., 2022), leverage scores, and gradient alignment analyses offer promising avenues for characterizing this balance, revealing whether model learning achieves a generalized understanding or relies disproportionately on memorized patterns (Fort et al., 2020; Chatterjee, 2020; Chatterjee & Zielinski, 2022; Zielinski et al., 2020).

Autoregressive models, useful for their promising extrapolation capabilities, accumulate errors during inference, in part due to the inevitable distribution shift that originates with the usage of model outputs as inputs to drive the predicted evolution arbitrarily into the future (Lee, 2023; Brandstetter et al., 2023). While such inference-time error accumulation is significant, we reveal a decisive learning limitation that also contributes to the difficulty of achieving stable long-term rollouts: gradient signals do not propagate coherently across time. Standard autoregressive training does not promote the learning of a mechanism for generalizing from supervision of one-step predictions to multi-frame dynamical evolution. Solution operator emulators must have properties that reflect the true data-generating mechanisms—the physics—rather

than merely exploiting empirical correlations among proximal points in feature space.

Current research increasingly emphasizes the development of PDE emulators designed to achieve robust and unified generalization across diverse physical scenarios (Sun et al., 2025; Ye et al., 2024; Herde et al., 2024; Subramanian et al., 2023). Contemporary PDE emulators employ a variety of architectures (Li et al., 2021; Lu et al., 2021; Gregory et al., 2024; Shankar et al., 2023); however, most large-scale deployments rely on UNet (Ronneberger et al., 2015) or Vision Transformer backbones (Vaswani et al., 2023; Liu et al., 2021; Dosovitskiy et al., 2021), and there remains no consensus on which model variant is most capable at scale. One must balance ease of optimization with the incorporation of physics priors, but quantitative tools for comparing loss-landscape properties across these architectures remain under-explored.

By quantifying gradient overlap between different training examples, it is possible to identify conflicts or synergies present during learning and inherent to the representational geometry of fully trained models. The sensitivity of model parameters to gradient updates derived from individual training samples is formalized through the use of influence functions (Hampel, 1974; Cook & Weisberg, 1982). Influence functions were originally developed in robust statistics (Huber & Ronchetti, 2009) to quantify how small perturbations of a data point in the training set affect model parameter estimations (Koh & Liang, 2020; Bae et al., 2022). Diagonal elements of the influence function measure each training example's self-leverage; high-leverage points thereby identify data that exerts disproportionate impact during training.

2. Related Work

Influence functions are powerful tools for understanding model behavior and data-attribution (Koh & Liang, 2020; Bae et al., 2022). Robust and interpretable criteria for detecting anomalous inputs follow from techniques that analyze the alignment of gradients (Wang et al., 2025) by quantifying directional consistency with in-distribution data (Huang et al., 2021), employ orthogonal projection (Behpour et al., 2023) to isolate anomalous components, and outlier gradient analysis (Chhabra et al., 2025).

(Fort et al., 2020) define stiffness in terms of the dot-product between the loss-gradients corresponding to two examples. Positive stiffness implies a stochastic gradient descent (SGD) step benefiting one example simultaneously lowers the loss of the other, evidence that the network assimilated shared, transferable features. Two summary statistics: sign-stiffness and cosine-stiffness, emphasize inter-class and intra-class correlations, respectively. Stiffness considered as a function of input-space distance yields a dynamic correlation length—the distance where average stiffness first crosses zero—which they found to shrink over epochs, revealing how the learned function becomes progressively more localized as specialization sets in.

The coherent gradients hypothesis (Chatterjee, 2020) proposed that per-example gradients tend to align for similar examples. Hence, SGD steps supported by many examples steer the network toward functions that generalize rather than memorize. Extensions of the coherent gradients hypothesis (Zielinski et al., 2020) posit that SGD updates aligned across multiple training examples (strong directions) underpin generalization, whereas idiosyncratic updates (weak directions) promote memorization. They introduce optimizers that suppress weak directions, reducing the train-test gap-even in the presence of heavy label noise. Complementing this view, He and Su (He & Su, 2020) establish the notion of local elasticity, wherein a parameter update perturbs predictions only within a narrow neighbourhood around the training point.

PINNfluence (Naujoks et al., 2024) interrogates a trained physics-informed neural network under perturbations to the PDE parameters and reweighting of collocation points. They distill raw pointwise influences into physically meaningful diagnostics such as the directional indicator, which measures the fraction of influence that propagates downstream with the fluid flow. In contrast to PINNfluence, we consider autoregressive vision models trained to generalize across distributions of initial conditions, rather than collocation-based function approximators fitted to a single PDE instantiation.

The dominant architectures for the image-to-image paradigm currently used in neural PDE emulation and PDE foundation modeling involve UNet and ViT backbones. Indeed, convolutional backbones derived from UNets remain standard architectures for grid-based surrogate modeling and large-scale benchmarking efforts (Takamoto et al., 2024; Gupta & Brandstetter, 2022). Likewise, transformer-based backbones now underpin many recent PDE foundation models and large-scale operator learners, including Poseidon and PDEformer (Herde et al., 2024; Ye et al., 2024), in addition to several contemporary multi-physics (Rautela et al., 2026) and shifting window transformer (Liu et al., 2021) architectures. Fourier neural operators provide an important complementary class of operator-learning models (Li et al., 2021; Subramanian et al., 2023); however, because their representations are constructed from truncated Fourier modes, the resulting plane-wave basis is less naturally suited to the spatially localized discontinuities and sharp shock fronts characteristic of compressible flow evolution (Ohana et al., 2025). Consistent with this expectation, we found FNO variants to be noncompetitive with both UNet and ViT backbones on the flows considered in this work. DeepONets provide a foundational operator-learning formalism (Lu et al.,

2021), but are not the prevailing architecture for contemporary large-scale PDE solution operator emulation. We note that architectures with built-in physics priors may show different coherence patterns by changing the local response geometry of the learned one-step map. However, such priors do not by themselves couple autoregressive training gradients across distinct time or class indices, and therefore do not explicitly provide a mechanism for increasing off-diagonal coherence.

3. Our Contributions

We extend influence functions (Koh & Liang, 2020; Bae et al., 2022), notions of stiffness (Fort et al., 2020), and the coherent gradients hypothesis (Chatterjee, 2020) from settings dominated by classification to neural emulators tasked with learning PDE solution operators. Our influence-based analysis permits a response-theoretic interpretation of gradient alignment through structural properties of the underlying differential operator.

Key advances contained in this work are enumerated below.

1. **Physical Time-Aware Analysis of The Influence Function:** A systematic study of off-diagonal influence-function elements in PDE surrogate models, capable of quantifying training-sample leverage across physical time. This diagnostic characterizes the coherence between gradient signals induced by examples of different time indices, revealing the extent to which learning dynamics reflect fundamental time-invariant structures of the underlying PDE.
2. **Gradient-Coherence Across Initial Condition Classes:** We determine the degree of alignment of gradients computed across different classes of PDE solutions for the two most frequently used architecture backbones, a UNet and a ViT. Strong alignment signals the learning of robust, transferable physics, whereas weak alignment suggests that the neural network embeds these classes on separated regions of the input manifold, with limited gradient coherence, despite the fact that the data represents solutions to the same underlying PDE.
3. **Physics-Informed Gradient Coherence:** We extend influence analysis by measuring the coherence between gradients derived from standard data-loss objectives and physics-informed cost functions, specifically global mass and global energy conservation. By evaluating cross-influence between conservation-law residuals and data-driven training losses, we quantify the degree to which scalable regression objectives induce compatible physical learning signals, providing a diagnostic for assessing whether the learning process supports robust operator-level generalization.
4. **Efficient and Exact Influence Function Computation:** We compute the hat-matrix metric-weighted gradient alignment by inverting the relevant Hessian-like susceptibility operator only on the mini-batch gradient-spanned subspace. While determining the parameter response requires inversion in the full parameter space, the loss sensitivity measured by the influence function depends only on metric-weighted inner-product of gradients. Hence, the inversion need be computed only in the low-rank gradient subspace, enabling exact and efficient measurement of cross-time and cross-class gradient coherence. In contrast to many influence analyses, which consider only the final trained model, this computationally efficient formulation permits evaluation across multiple training checkpoints, thereby enabling direct study of training dynamics.
5. **Dynamical Proximal Influence Formalism:** We formulate influence functions through a proximal objective that induces a Riemannian metric determined by the hat matrix of robust statistics. This extends influence analysis from classification tasks to regression and reinterprets influence as the Lie derivative of the cost along the training-flow vector field. Consequently, influence is promoted from a static sensitivity measure defined only at optimality to a dynamical observable that can be interpreted throughout training.

This paper is organized as follows. First, model architectures, datasets, and optimization procedures are described in section 4. We then develop the mathematical formulation of the proximal influence-response framework in section 5. Finally, our influence-analysis results and discussion are presented in section 6, exposing limitations in the generalization capabilities of autoregressively trained PDE emulators.

4. Data and Training

We trained neural network surrogate models to approximate the solution operators of the two-dimensional compressible Euler (CE) and Navier-Stokes (NS) equations using data from PDEGym (Herde et al., 2024), in addition to the one-dimensional Burgers equation from PDEBench (Takamoto et al., 2024). To learn CE evolution, we used three datasets for training, namely those containing initial conditions for the four quadrant Riemann problem with (CE-RPUI) and without (CE-RP) uncertain interfaces, in addition to the curved Riemann problem (CE-CRP). For NS flows, we also used three datasets for training, namely, those containing Brownian Bridge (NS-BB), Gaussian random field (NS-Gauss), and sinusoidal (NS-Sines) initial conditions. Training on the velocity fields corresponding to solutions of the NS equations [see Figure 8] involves a feature space distinct from CE data and flow morphology,

involving smoother, viscosity-regularized transport with vorticity-dominated structure. On the Burgers equation data, we considered solutions of small viscosity 10^{-3}, so that the resulting time evolution develops well-resolved shock waves.

Since instantaneous flow states alone cannot distinguish viscous NS flows from their inviscid CE counterparts, we do not combine CE data with NS data. Furthermore, rather than representing a fluid state using the velocities and pressure in addition to density, as was done by Poseidon (Herde et al., 2024), we used the momentum and energy fields; we expect that this setup will better facilitate the learning of all four conservation laws.

Each snapshot of the flow state at discrete time t is represented as a set of four spatially discretized fields $\rho_{\text{mass}}, \rho^i_{\text{mom}}, \rho_{\text{energy}}$ on a uniform grid of size 128×128, where ρ_{mass} denotes mass density, ρ^i_{mom} are the Cartesian components of momentum density, and ρ_{energy} is energy density. The model $\hat{y}_\theta$ is trained to emulate the compressible Euler evolution, i.e., $\hat{y}_\theta \approx U$, where the operator U enacts

$$s_{t+1} \;=\; U[s_t], \tag{1}$$

with s_t the collection of state variables at timestep t, via optimization of the model weights θ. Specifically, we used the Adam optimizer (Kingma & Ba, 2017) with learning rate 5×10^{-4} and weight decay $\lambda = 10^{-4}$ to minimize a scaled mean squared error (SMSE) between predicted and true states

$$C_{\text{SMSE}}(\theta) = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{4}\sum_{c}\frac{||\hat{y}^c_\theta(s_{t_n}) - s^c_{t_n+1}||^2_{L_2}}{\text{RMS}(s^c_{t_n+1})}, \tag{2}$$

on mini-batches containing $N = 48$ transitions $s_t \to s_{t+1}$, chosen randomly from the training set; here, the L_2 norm is computed over the spatial degrees of freedom and the channel index c runs through mass, both cartesian components of the momentum, and energy, respectively. $\text{RMS}(s^c)$ is computed channel-wise by taking the spatial root-mean-square of s^c. Thus, Equation 2 strikes a balance between ordinary and relative mean-squared-error; normalizing each channel's squared error by the target fields' characteristic amplitude favors examples containing pronounced features, but does not completely drown out gradients derived from relatively quiescent flows, thereby facilitating accurate capture of high-energy shocks and wavefronts without harsh under-emphasis of small-amplitude features. Moreover, the scaling of Equation 2 renders dimensionless the matrix elements of interest to this work.

In order to compare the results of our experiments across model architectures, we trained both a UNet (Ronneberger et al., 2015) and a vision transformer (ViT) (Vaswani et al., 2023; Liu et al., 2021; Dosovitskiy et al., 2021). Our UNet was based on BigGAN (Brock et al., 2019), with four downsampling blocks and 24 channels after the initial embedding layer, for a total of about 13-million parameters. Our ViT was of layer-depth six, with 256 channels, for a total of about 5-million parameters, fewer than our UNet due to memory constraints on the 40 GB A100s that we used for training. The validation losses for each of our CE-tasked models is shown in Figure 7; for NS loss curves, see Figure 8. Each of the two architectures was trained three times, sharing those three seeds that controlled initialization and dataset split. The training of each model was performed in distributed mode across two such A100s.

For remarks on data selection see Appendix A.

5. Proximal Response Function

We develop the influence function as follows, taking inspiration from (Bae et al., 2022). Let θ be the current parameter values and consider the optimization step $\theta \leftarrow \theta + \delta\theta$, where the tangent-space displacement $\delta\theta$ minimizes the proximal objective

$$S(\delta\theta) = dC[\delta\theta] + \frac{1}{2}||\hat{y}(\theta) - \hat{y}(\theta + \delta\theta)||^2_{L_2}, \tag{3}$$

with $\hat{y}$ a neural network. The stationarity condition $dS \overset{!}{=} 0$ is satisfied (to linear order) by

$$\delta\theta^\mu = -\,\eta^{\mu\nu}\partial_\nu C, \tag{4}$$

where $\eta_{\mu\nu} = J^n_\mu J^n_\nu$ is the hat-matrix metric, $J^n_\mu = \partial_\mu \hat{y}^n$ and is the model Jacobian. Throughout, Roman indices label mini-batch examples, while Greek indices label model parameters. By convention, the components of η carry lowered indices, $\eta_{\mu\nu}$, while those of η^{-1} carry raised indices, $\eta^{\mu\nu}$, i.e. $\eta^{\mu\alpha}\eta_{\alpha\nu} = \delta^\mu{}_\nu$; η provides the canonical correspondence between covariant and contravariant components (Absil et al., 2008). Unless otherwise specified, Greek indices obey Einstein summation notation. Equation 3 balances the force term dC against the kinetic cost of the update distance in the η prescribed geometry. Convexity of η, together with mild regularity requirements on C, guarantees a unique stationary point of each proximal subproblem. Proximal gradient descent iterates these subproblems to accumulate a sequence of locally improving displacements that drives descent of cost function C. The inverse susceptibility tensor η^{-1} serves as a generalized stiffness operator by propagating gradient signals to parameter displacements (Fort et al., 2020). For an intuitive derivation of the influence function, see Appendix B

5.1. Dynamical Proximal Influence Formalism

Classical influence functions can be expressed as the Lie derivative of a scalar; they're capable of probing local gradient coherence, generalization capabilities, and adversarial

sensitivity, in addition to enabling the identification of high-leverage examples. Consider a scalar observable Q and a vector field $V = -\eta^{-1}(dC)$ derived from the proximal objective. The Lie derivative of Q along V is

$$\begin{aligned}\mathcal{L}_V Q &= (\partial_\mu Q)\ \eta^{\mu\nu}\ (-\partial_\nu C) \\ &= -\left(\frac{\delta Q}{\delta \hat{y}^n}, \Pi^{nm}\frac{\delta C}{\delta \hat{y}^m}\right),\end{aligned} \tag{5}$$

where

$$\Pi^{nm} = J^n_\mu \eta^{\mu\nu} J^m_\nu, \tag{6}$$

and the inner product $(\cdot,\cdot)$ is performed over feature indices. Hence, when Q is a loss function, Equation 5 reduces to the familiar form of an influence function in deep learning: the infinitesimal response of the loss, expressible as a metric-weighted gradient overlap. Likewise, when Q denotes a model response and V encodes the perturbation to the gradient signal induced by a deformation of the input, Equation 5 reproduces the classical influence-function expression from robust statistics [see Appendix C].

Evidently, the Lie-derivative formulation of response is well defined at any point along the training trajectory, as it depends only on the instantaneous training-flow vector field and the induced local geometry; hence, influence-function analysis of neural networks does not require attainment of a stationary point. This perspective elevates influence from a static sensitivity relevant only near convergence to a dynamical linear response observable defined throughout optimization. Note that the language of differential geometry is natural for discussing transport in non-Euclidean metric spaces. Training proceeds along a flow, and observable evolution is expressed by the Lie derivative, whose defining input is a vector field rather than an isolated local tangent direction. This distinction is necessary for mathematical precision and computational correctness: the metric identifies loss differentials with parameter-space vector fields, thereby providing the well-known dynamical structure that justifies our interpretation of the influence response along the optimization trajectory.

As Π assumes the form of a classical hat-matrix, its diagonal elements reflect statistical leverage, quantifying the self-influence of individual training examples, while off-diagonals measure cross-influence, i.e. influence between distinct examples. High leverage scores identify regions of parameter space with strong local curvature or limited redundancy, i.e., points with disproportionately large influence on the global response structure. Furthermore, the response matrix encodes the pairwise overlap of example gradients — effectively probing the local loss landscape by highlighting directions of correlated curvature and shared descent paths. The response matrix Π^{nm} tracks how gradients derived from each output feature of prediction m influence each output feature of prediction n, offering an investigative level of detail across spacetime, channel, and class dimensions that remains unexplored. To avoid materializing Π, which has $(128\times128\times4\times B)^2$ elements, where B is the mini-batch size, we consider coarse observables: scaled mean squared error (SMSE), in addition to global mass and energy conservation.

5.2. Observables

Our probe of generalization capabilities quantifies the coherence of gradients derived from test data cost functions of physical and statistical significance. We introduce three generalized residuals

$$r_{\mathrm{C}} = \frac{\delta C_{\mathrm{SMSE}}}{\delta \hat{y}_\theta} = \frac{1}{4}\sum_c \frac{\hat{y}^c_\theta - y^c}{\mathrm{RMS}(y^c)}, \tag{7a}$$

$$r_{\mathrm{M}} = \frac{\delta C_{\mathrm{Mass}}}{\delta \hat{y}_\theta} = \frac{M[\hat{y}_\theta] - M[x]}{M[x]}, \tag{7b}$$

$$r_{\mathrm{E}} = \frac{\delta C_{\mathrm{Energy}}}{\delta \hat{y}_\theta} = \frac{E[\hat{y}_\theta] - E[x]}{E[x]}, \tag{7c}$$

where M (E) computes the total mass (energy) of its argument, x is the input state that evolves to y, i.e. $y = U[x]$, where U is defined in Equation 1, and $\hat{y}$ is the neural network approximation to y. Recall that each training example is comprised of pairs $(x, y) = (s^n_t,\ s^n_{t+1})$ of states s sharing a common initial configuration indexed by n, and related by the compressible Euler evolution operator. Viewed as a coupling matrix over residuals, the diagonal blocks of Π recover the usual influence (e.g., how a perturbation in the SMSE affects SMSE itself), while the off-diagonal blocks encode cross-coupling, quantifying how a change in the SMSE residual at one time step or sample is converted into a physics error at another time step or sample, and vice versa.

It is useful to introduce following notation for the remaining external indices of the response matrix

$$H^{tn\,\tau m}_{AB} = \left(r^{tn}_A, \Pi^{tn\,\tau m} r^{\tau m}_B\right) = g^{tn}_{A,\,\mu}\eta^{\mu\nu} g^{\tau m}_{B,\,\nu}\,, \tag{8}$$

where n, m index trajectories, defined by distinct initial conditions, indices t and τ specify the time step along said trajectories, and parameter gradients g are derived from their respective cost functions C. Hence, H_{CC} gives the change in SMSE due to an SMSE perturbation, while H_{MC} and H_{EC} propagate the effect of gradients derived from SMSE into the physics informed mass and energy conservation errors, respectively.

For CE and NS data, matrix elements of Π were determined for 18 different mini-batches, each of which contained 3 trajectories corresponding to distinct initial conditions over 16 time-steps, for each of the 3 seeds from which each model architecture was trained; on Burgers data, we included 32 time-steps per trajectory.

In practice, we include an ℓ_2-type regularizer corresponding to the weight-decay term used during AdamW optimization (Kingma & Ba, 2017; Loshchilov & Hutter, 2019), $\eta_{\mu\nu} \to \eta_{\mu\nu} + \lambda\delta_{\mu\nu}$, with λ weakly lifting the zero modes of η. Empirically, our results do not change if λ is chosen to be either 10^{-6} or 10^{-4}. These mini-batches match the training batch size, and all reported influence summaries are aggregated over time-index, class-index, mini-batch, and model seed, with range bars showing the empirical variability. This variability is sufficiently small compared with the observed separation between diagonal and off-diagonal response, between temporally local and nonlocal response, and between intra-class and inter-class response; thus the reported coherence patterns are statistically resolved, and the corresponding conclusions are not expected to change under additional mini-batch sampling.

As a sensitivity measure, the influence function possesses no canonical scale. Common choices include normalization by gradient magnitude, yielding the cosine similarity, normalization by trace or Frobenius norms, standardized influence functions (Héritier & Ronchetti, 1994; Lu et al., 1997), or alternatively leaving the response unnormalized altogether. In the experiments below, we use cosine normalization, reporting influence after division by both the perturbation and response gradient magnitudes. This normalization sets each self-response to unity, which removes the absolute scale of local sensitivities and hence isolates the cross-response magnitude, a comparison appropriate when the effects of influence are accumulated over many optimization steps.

5.3. Efficiently Computing the Influence Function

Influence-function computations in deep learning are commonly performed either through iterative matrix-free solvers, such as conjugate-gradient methods (Orban & Arioli, 2017; Montoison & Orban, 2023), or through scalable approximations to the curvature operator (George, 2021; TransferLab, 2024). While such approaches are often necessary in high-dimensional parameter spaces, iterative methods require repeated Hessian-vector products and convergence tolerances, whereas generic low-rank or diagonal approximations do not provide controlled error estimates.

In the present setting, however, the influence observable depends only on metric-weighted overlaps between example gradients. Let $G^n_\alpha = \partial_\alpha C^n$ denote the gradient associated with mini-batch example n. Performing a QR decomposition on G yields an orthonormal basis Q that projects onto the gradient subspace. Hence, $H^{nm} = g^n_\mu Q^\mu_a \hat{\eta}^{ab} Q^\nu_b g^m_\nu$, where $\hat{\eta}_{ab} = Q^\mu_a \eta_{\mu\nu} Q^\nu_b$ can be efficiently inverted. Since the dimension of the gradient-spanned subspace is bounded by the mini-batch size rather than the total number of trainable parameters, the resulting inversion is both exact and computationally efficient, eliminating the need for uncontrolled curvature approximations or iterative parameter-space solves.

We emphasize that since both external covectors entering the response observable are elements of the batch gradient-spanned subspace, directions orthogonal to it are never probed by the bilinear form; the computation is therefore exact for the measured influence matrix, rather than a low-rank or approximate realization of an otherwise evaluated full parameter-space displacement.

6. Results

We examine how gradient information propagates across time and initial-condition classes in autoregressive PDE emulators using influence-based diagnostics evaluated on held-out test data. Across architectures, physical observables, and datasets, the measured response geometry exhibits localization in both temporal and class directions, with off-diagonal influence decaying as temporal separation increases. The resulting response structure indicates limited gradient coherence, consistent with a short effective correlation length in feature space, indicating that these models primarily learn time- and class-indexed update rules rather than a globally consistent dynamical operator.

6.1. Physical Time-Aware Analysis of Influence

Measurements of the two-time influence function [see Equation 8] for both a UNet and a ViT tasked with emulating fluid flow exhibit rapid temporal decay in the off-diagonal terms [see Figure 1], which indicates that surrogate training constructs localized vector fields suitable only for interpolation within small neighborhoods of the training data. Ideally, learning the solution operator to a PDE that lacks explicit time dependence would involve constructive coupling of gradients derived from examples at a different times, for the dynamics are time-translation equivariant.

We emphasize that training pairs drawn from solutions to an autonomous PDE correspond to the same time-invariant solution operator evaluated at different points along a trajectory. This provides the justification for using a one-step training scheme, because gradients at different times pose the same learning problem to the model parameters. Hence, marked off-diagonal decay in the two-time influence reflects undesirable weak coupling between gradient signals that are generated by identical operator instances.

The response geometry also correlates with inference-time degradation [see Table 2]; examples exhibiting stronger cross-time response tend to exhibit lower rollout error, indicating that H_{CC} contains predictive information about autoregressive error accumulation.

Table 1. Correlation between cross-time influence and cumulative rollout error as a function of extrapolation time on CE trained models. For NS data companion figure, see Table 3.

Rollout Time	1	2	3	4	5	6	7	8	9	10
ViT	-0.67	-0.67	-0.67	-0.66	-0.65	-0.64	-0.63	-0.63	-0.62	-0.62
UNet	-0.59	-0.57	-0.53	-0.50	-0.48	-0.45	-0.42	-0.40	-0.37	-0.35

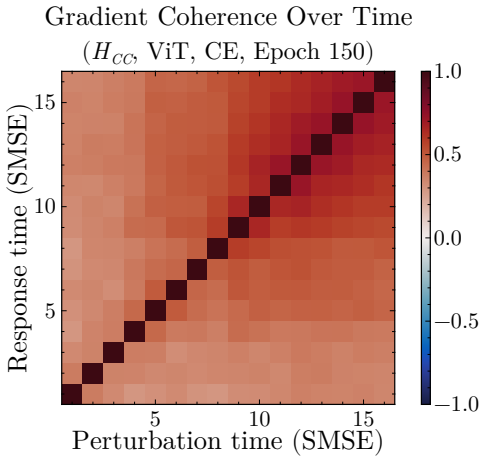

Figure 1. Heatmap of two-time influence (H_{CC}) for our ViTs trained on CE data, shown as a function of perturbation time (horizontal axis) and response time (vertical axis). Each pixel reports the intra-class averaged response induced by test example gradients at the perturbation time. A narrow diagonal ridge corresponds to time-local sensitivity consistent with interpolation, rather than generalization; substantial off-diagonal structure would indicate time-transferable learning. For the analogous plot using our UNets, see Figure 14. For NS data counterpart figure, see Figure 15 and Figure 16.

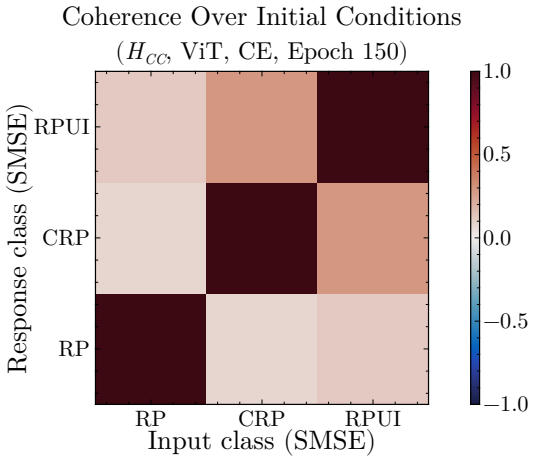

Figure 2. Class-to-class transferability matrix for our ViTs trained on the three-class CE split labeled RP, CRP, and RPUI. Each entry reports the time-averaged influence (H_{CC}) of test examples from the input class (horizontal axis) on examples from the response class (vertical axis). Diagonal dominance indicates class-locked gradient geometry; substantial off-diagonal values would imply shared learning across classes. For the analogous plot using our UNets, see Figure 10. For NS data counterparts, see Figure 11 and Figure 12.

6.2. Gradient-Coherence Across Classes

We find that the class-to-class transferability matrix [see Figure 2] is largely diagonal, indicating that gradient geometry is effectively class-locked: updates supported by one initial-condition family produce negligible response in the others. Furthermore, there is a minimal inter-class influence across time [see Figure 3].

6.3. Influence dependence on feature space distance

While all inputs to the network are related as unique solutions to a shared equation of motion under different initial conditions, both our ViTs and our UNets render CE examples well separated in the sense that the corresponding gradients don't significantly overlap unless the feature space distance between examples is small [see Figure 4], suggesting that training does not select a basin of convergence supportive of sharing information across examples during learning. Our results reveal a central limitation of PDE emulators, as it demonstrates that physics-agnostic training these models is prone to effectively treating different flow fields as distinct, isolated learning tasks. That this happens even when said classes of solutions arise merely from different initial conditions to the same physical process underscores the need for inductive biases to be explicitly incorporated.

Because NS trajectories occupy a narrower feature-space range, varying less strongly across time and initial-condition classes than CE, the measured response on this data exhibits a longer correlation length [see Figure 31], with off-diagonal sectors larger in magnitude [see Figure 39]. Thus our diagnostic is not architecture bound; it is largely dataset geometry that controls cross-time and cross-class gradient coherence, supporting a data-geometric rather than model-dependent interpretation.

6.4. Physics-Informed Gradient Coherence

In addition to cost-function gradient overlap, we also considered gradients derived from physics-informed diagnostics, namely global mass conservation [see Figure 19] and global energy conservation [see Figure 25]. The response matrices

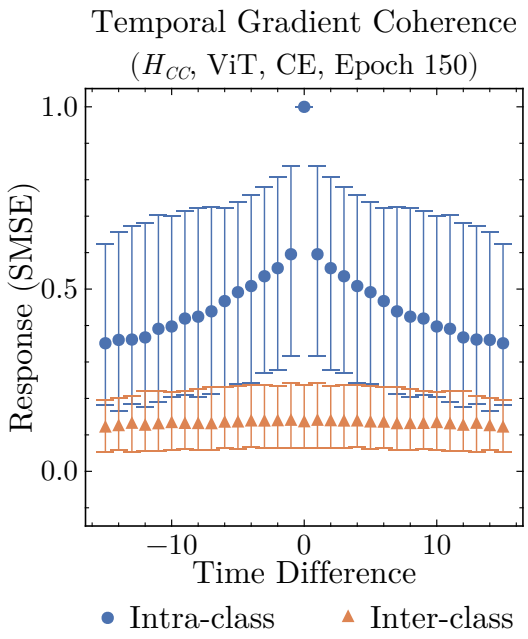

Figure 3. Time-lag summary of temporal transferability for our ViTs on CE data. The influence (H_{CC}) is averaged over all time-pairs of the same time difference and then split into intra-class pairs (gradient and response drawn from the same initial-condition class) and inter-class pairs (distinct initial-condition classes). Strong concentration near zero time difference indicates that gradient information fails to propagate coherently across time, while the lack off inter-class influence reveals an absence of physics consistent generalization. For the analogous plot using our UNets, see Figure 18. For NS data counterparts, see Figure 21 and Figure 22.

H_{MC} and H_{EC} exhibit suppressed cross-objective overlap between SMSE gradients and conservation-law gradients, with normalized influence of order 0.1. Thus, although the conservation gradients are not orthogonal to the data-fitting gradient geometry, they enter only as weakly coupled directions relative to the SMSE self-response. The corresponding time-lag summaries [see Figure 23 and Figure 27] show that this weak cross-objective coupling is coherent only within an initial-condition class, mirroring the class-local response geometry observed for H_{CC}.

6.5. Influence Function Dynamics

Training dynamics of the response geometry are probed by tracking off-diagonal influence elements across checkpointed epochs [see Figure 5]. The off-diagonal component measures gradient coupling across separated times and distinct initial-condition classes. Throughout CE training, the off-diagonal influence contributions for both UNet and ViT remain approximately constant at values not typically exceeding 0.3, indicating that learning proceeds in a response geometry biased toward localized intra-class coupling rather than marked cross-time or cross-class coherence.

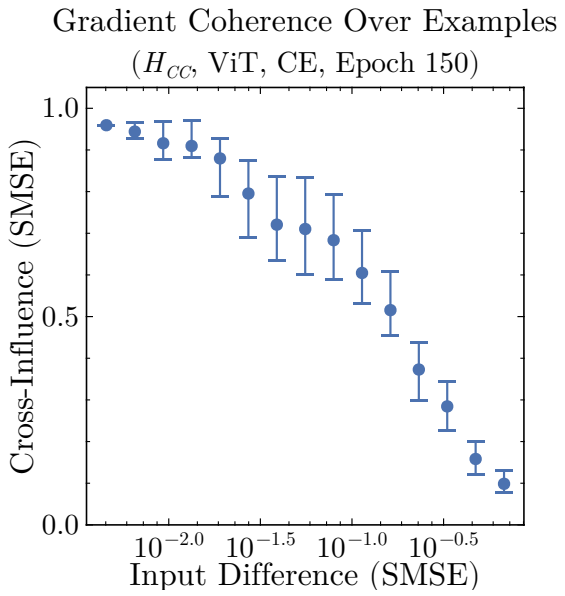

Figure 4. Influence as a function of feature-space separation between input states for CE data, comparing UNet and ViT; rangebars show variability across seeds. A steep decay indicates short-range interpolation of influence across the learned data manifold, implying that parameter updates affect only nearby states and generalization is limited. For NS data counterpart, see Figure 32.

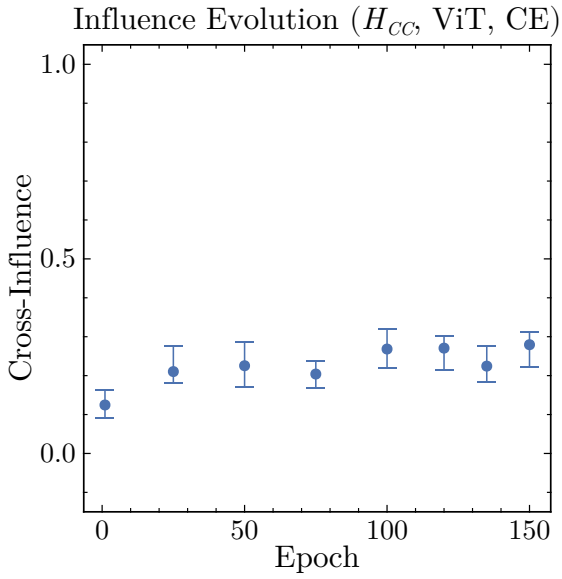

Figure 5. Off-diagonal components of the two-time response across checkpointed training epochs. The off-diagonal response remains approximately constant throughout optimization, indicating that cross-time gradient coupling is established early and does not substantially reorganize during training. For the corresponding UNet curves, see Figure 38.

On NS data, the off-diagonal response is likewise approximately constant across time pairs, but at stronger values, with normalized influence near 0.4. Influence observables

corresponding to the main-text figures are evaluated across checkpointed training epochs. Representative dynamical counterparts of the main-text diagnostics are deferred to Appendix D, where the evolution of the corresponding response geometry is visualized.

6.6. Influence Dependence on Data Dimensionality

The one-dimensional Burgers problem provides a simpler task than PDEGym; this difference is reflected in the response geometry [see Figure 6]. Early in training, the two-time response remains strongly localized near the temporal diagonal. As optimization proceeds, the off-diagonal response is progressively lifted, indicating the emergence of longer-range temporal gradient coherence. The resulting response matrix presents off-diagonal support at both early and late times: early-time states share features because of periodic initializations, whereas late-time states become similar through viscous smoothing.

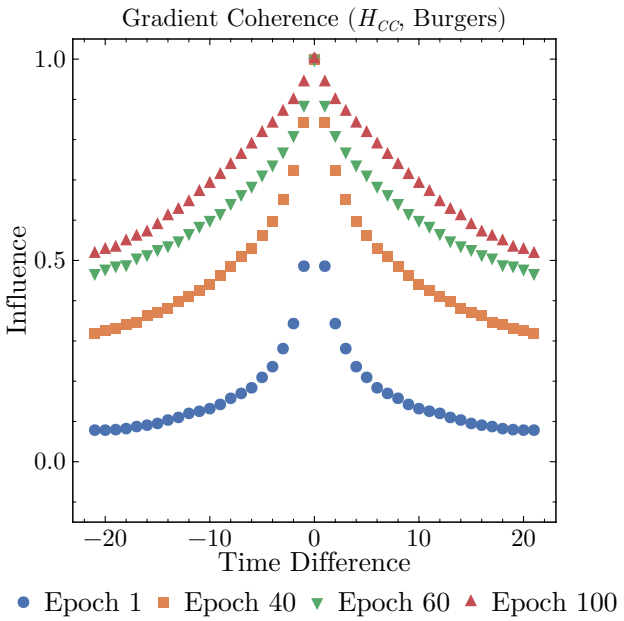

Figure 6. Time-lag summary of temporal transferability for the ViT trained on 1D Burgers data. The influence is averaged over all time-pairs with fixed time difference. Curves correspond to distinct training checkpoints, thereby resolving the evolution of the local response geometry during optimization.

7. Limitations

Our analysis is restricted to two widely used architectural backbones, UNets and ViTs. While these models represent dominant paradigms in contemporary large-scale PDE emulation, they do not exhaust the space of operator-learning methods. Consequently, the degree to which our measured gradient coherence properties depend on the chosen backbone rather than on the autoregressive training paradigm itself remains only partially resolved.

A second limitation is that we consider only standard one-step autoregressive training, without mechanisms explicitly designed to couple gradients across time. Recent work has explored approaches such as noise-conditioned trajectory perturbations, pushforward tricks, scheduled sampling, and multi-step rollout objectives. Such methods may enhance off-diagonal temporal coherence by preventing gradients from remaining localized near the training-time diagonal.

8. Conclusion

We reveal a shortcoming of physics-agnostic PDE emulators: it is not an immediate consequence of autoregressive training that the resulting model can satisfy those stringent expectations that follow from the governing equations one is trying to emulate. Physical considerations demand cooperative learning across trajectories, yet our results reveal lackluster support of off-diagonal response, suggesting limited learning of transferable physics, a necessary ingredient for robust generalization. In summary, we introduce targetable characteristics—time and class aware cross-influence—to guide research on algorithms for learning the underlying physics of a PDE and achieving robust long-horizon predictions across varied initial conditions.

Software and Data

We trained our models on the openly available dataset PDEGym (Herde et al., 2024) and PDEBench (Takamoto et al., 2024) using Lux.jl (Pal, 2023b;a), with Zygote.jl as our auto-differentiation backend (Innes, 2018). Plots in this manuscript were generated using Makie.jl (Danisch & Krumbiegel, 2021).

The code used in this work is available at https://github.com/lanl/PDEHats.

Acknowledgements

Research presented in this report was supported by the Laboratory Directed Research and Development program of Los Alamos National Laboratory under project number(s) 20250637DI, 20250638DI, and 20250639DI. This research used resources provided by the Los Alamos National Laboratory Institutional Computing Program, which is supported by the U.S. Department of Energy National Nuclear Security Administration under Contract No. 89233218CNA000001. It is published under LA-UR-25-28084.

Impact Statement

This work contributes a diagnostic framework for distinguishing mere memorization from genuine generalization in autoregressive surrogate models, with particular relevance to scientific and engineering applications where model failures can have downstream consequences. By exposing training-dynamics limitations that are not visible through standard accuracy metrics, our analysis helps identify when learned surrogates are likely to be brittle under long-horizon rollout or distribution shift, informing safer deployment in settings such as climate modeling, fluid dynamics, and materials simulation. The methods introduced here are diagnostic rather than prescriptive and do not directly enable new capabilities for misuse; instead, they promote transparency and reliability by clarifying when and why models fail to learning shared physical structure. More broadly, our research encourages the development of learning algorithms and evaluation practices that prioritize robustness and interpretability, supporting the responsible use of machine learning in high-consequence scientific workflows.

References

Absil, P.-A., Mahony, R., and Sepulchre, R. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, 2008. ISBN 978-0-691-13298-3.

Bae, J., Ng, N., Lo, A., Ghassemi, M., and Grosse, R. If influence functions are the answer, then what is the question?, 2022. URL https://arxiv.org/abs/2209.05364.

Batatia, I., Benner, P., Chiang, Y., Elena, A. M., Kovács, D. P., Riebesell, J., Advincula, X. R., Asta, M., Avaylon, M., Baldwin, W. J., Berger, F., Bernstein, N., Bhowmik, A., Blau, S. M., Cărare, V., Darby, J. P., De, S., Pia, F. D., Deringer, V. L., Elijošius, R., El-Machachi, Z., Falcioni, F., Fako, E., Ferrari, A. C., Genreith-Schriever, A., George, J., Goodall, R. E. A., Grey, C. P., Grigorev, P., Han, S., Handley, W., Heenen, H. H., Hermansson, K., Holm, C., Jaafar, J., Hofmann, S., Jakob, K. S., Jung, H., Kapil, V., Kaplan, A. D., Karimitari, N., Kermode, J. R., Kroupa, N., Kullgren, J., Kuner, M. C., Kuryla, D., Liepuoniute, G., Margraf, J. T., Magdău, I.-B., Michaelides, A., Moore, J. H., Naik, A. A., Niblett, S. P., Norwood, S. W., O'Neill, N., Ortner, C., Persson, K. A., Reuter, K., Rosen, A. S., Schaaf, L. L., Schran, C., Shi, B. X., Sivonxay, E., Stenczel, T. K., Svahn, V., Sutton, C., Swinburne, T. D., Tilly, J., van der Oord, C., Varga-Umbrich, E., Vegge, T., Vondrák, M., Wang, Y., Witt, W. C., Zills, F., and Csányi, G. A foundation model for atomistic materials chemistry, 2024. URL https://arxiv.org/abs/2401.00096.

Behpour, S., Doan, T., Li, X., He, W., Gou, L., and Ren, L. Gradorth: A simple yet efficient out-of-distribution detection with orthogonal projection of gradients, 2023. URL https://arxiv.org/abs/2308.00310.

Bodnar, C., Bruinsma, W. P., Lucic, A., Stanley, M., Vaughan, A., Brandstetter, J., Garvan, P., Riechert, M., Weyn, J. A., Dong, H., Gupta, J. K., Thambiratnam, K., Archibald, A. T., Wu, C.-C., Heider, E., Welling, M., Turner, R. E., and Perdikaris, P. A foundation model for the earth system, 2024. URL https://arxiv.org/abs/2405.13063.

Brandstetter, J., Worrall, D., and Welling, M. Message passing neural pde solvers, 2023. URL https://arxiv.org/abs/2202.03376.

Brock, A., Donahue, J., and Simonyan, K. Large scale gan training for high fidelity natural image synthesis, 2019. URL https://arxiv.org/abs/1809.11096.

Chatterjee, S. Coherent gradients: An approach to understanding generalization in gradient descent-based optimization, 2020. URL https://arxiv.org/abs/2002.10657.

Chatterjee, S. and Zielinski, P. On the generalization mystery in deep learning, 2022. URL https://arxiv.org/abs/2203.10036.

Chhabra, A., Li, B., Chen, J., Mohapatra, P., and Liu, H. Outlier gradient analysis: Efficiently identifying detrimental

training samples for deep learning models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=v77ZMzbsBA.

Cook, R. D. and Weisberg, S. *Residuals and Influence in Regression*. Chapman and Hall, New York, 1982. ISBN 0-412-24280-0.

Danisch, S. and Krumbiegel, J. Makie.jl: Flexible high-performance data visualization for Julia. *Journal of Open Source Software*, 6(65):3349, 2021. doi: 10.21105/joss.03349. URL https://doi.org/10.21105/joss.03349.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale, 2021. URL https://arxiv.org/abs/2010.11929.

Fort, S., Nowak, P. K., Jastrzebski, S., and Narayanan, S. Stiffness: A new perspective on generalization in neural networks, 2020. URL https://arxiv.org/abs/1901.09491.

George, T. NNGeometry: Easy and Fast Fisher Information Matrices and Neural Tangent Kernels in PyTorch, February 2021. URL https://doi.org/10.5281/zenodo.4532597.

Gregory, W. G., Hogg, D. W., Blum-Smith, B., Arias, M. T., Wong, K. W. K., and Villar, S. Equivariant geometric convolutions for emulation of dynamical systems, 2024. URL https://arxiv.org/abs/2305.12585.

Gupta, J. K. and Brandstetter, J. Towards multi-spatiotemporal-scale generalized pde modeling, 2022. URL https://arxiv.org/abs/2209.15616.

Hampel, F. R. The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, 69(346):383–393, 1974. doi: 10.1080/01621459.1974.10482962.

He, H. and Su, W. J. The local elasticity of neural networks, 2020. URL https://arxiv.org/abs/1910.06943.

Herde, M., Raonić, B., Rohner, T., Käppeli, R., Molinaro, R., de Bézenac, E., and Mishra, S. Poseidon: Efficient foundation models for pdes, 2024. URL https://arxiv.org/abs/2405.19101.

Héritier, S. and Ronchetti, E. Robust bounded-influence tests in general parametric models. *Journal of the American Statistical Association*, 89(427):897–904, 1994. ISSN 0162-1459. URL https://www.jstor.org/stable/2290914.

Huang, R., Geng, A., and Li, Y. On the importance of gradients for detecting distributional shifts in the wild, 2021. URL https://arxiv.org/abs/2110.00218.

Huber, P. J. and Ronchetti, E. M. *Robust Statistics*. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc., 2009. ISBN 9780470129906. doi: 10.1002/9780470434697. URL https://doi.org/10.1002/9780470434697.

Innes, M. Don't unroll adjoint: Differentiating ssa-form programs. *CoRR*, abs/1810.07951, 2018. URL http://arxiv.org/abs/1810.07951.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization, 2017. URL https://arxiv.org/abs/1412.6980.

Koh, P. W. and Liang, P. Understanding black-box predictions via influence functions, 2020. URL https://arxiv.org/abs/1703.04730.

Krishnapriyan, A. S., Gholami, A., Zhe, S., Kirby, R. M., and Mahoney, M. W. Characterizing possible failure modes in physics-informed neural networks, 2021. URL https://arxiv.org/abs/2109.01050.

Lee, Y. Autoregressive renaissance in neural pde solvers. In *ICLR Blogposts 2023*, 2023. URL https://iclr-blogposts.github.io/2023/blog/2023/autoregressive-neural-pde-solver/.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A. Fourier neural operator for parametric partial differential equations, 2021. URL https://arxiv.org/abs/2010.08895.

Lippe, P., Veeling, B. S., Perdikaris, P., Turner, R. E., and Brandstetter, J. Pde-refiner: Achieving accurate long rollouts with neural pde solvers, 2023. URL https://arxiv.org/abs/2308.05732.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows, 2021. URL https://arxiv.org/abs/2103.14030.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

Lu, J., Ko, D., and Chang, T. The standardized influence matrix and its applications. *Journal of the American Statistical Association*, 92(440):1572–1580, 1997. ISSN 0162-1459. URL https://www.jstor.org/stable/2965428.

Lu, L., Jin, P., Pang, G., Zhang, Z., and Karniadakis, G. E. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, March 2021. ISSN 2522-5839. doi: 10.1038/s42256-021-00302-5. URL http://dx.doi.org/10.1038/s42256-021-00302-5.

Montoison, A. and Orban, D. Krylov.jl: A Julia basket of hand-picked Krylov methods. *Journal of Open Source Software*, 8(89):5187, 2023. doi: 10.21105/joss.05187.

Naujoks, J. R., Krasowski, A., Weckbecker, M., Wiegand, T., Lapuschkin, S., Samek, W., and Klausen, R. P. Pinnfluence: Influence functions for physics-informed neural networks, 2024. URL https://arxiv.org/abs/2409.08958.

Ohana, R., McCabe, M., Meyer, L., Morel, R., Agocs, F. J., Beneitez, M., Berger, M., Burkhart, B., Burns, K., Dalziel, S. B., Fielding, D. B., Fortunato, D., Goldberg, J. A., Hirashima, K., Jiang, Y.-F., Kerswell, R. R., Maddu, S., Miller, J., Mukhopadhyay, P., Nixon, S. S., Shen, J., Watteaux, R., Blancard, B. R.-S., Rozet, F., Parker, L. H., Cranmer, M., and Ho, S. The well: a large-scale collection of diverse physics simulations for machine learning, 2025. URL https://arxiv.org/abs/2412.00568.

Orban, D. and Arioli, M. *Iterative Solution of Symmetric Quasi-Definite Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2017. doi: 10.1137/1.9781611974737. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611974737.

Pal, A. On Efficient Training & Inference of Neural Differential Equations, 2023a.

Pal, A. Lux: Explicit Parameterization of Deep Neural Networks in Julia, April 2023b. URL https://doi.org/10.5281/zenodo.7808903.

Raissi, M., Perdikaris, P., and Karniadakis, G. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2018.10.045.

Rautela, M. S., Most, A., Mansingh, S., Love, B. C., Scheinker, A., Oyen, D., Debardeleben, N., Lawrence, E., and Biswas, A. Morph: Pde foundation models with arbitrary data modality, 2026. URL https://arxiv.org/abs/2509.21670.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation, 2015. URL https://arxiv.org/abs/1505.04597.

Shankar, V., Barwey, S., Kolter, Z., Maulik, R., and Viswanathan, V. Importance of equivariant and invariant symmetries for fluid flow modeling, 2023. URL https://arxiv.org/abs/2307.05486.

Subramanian, S., Harrington, P., Keutzer, K., Bhimji, W., Morozov, D., Mahoney, M., and Gholami, A. Towards foundation models for scientific machine learning: Characterizing scaling and transfer behavior, 2023. URL https://arxiv.org/abs/2306.00258.

Sun, J., Liu, Y., Zhang, Z., and Schaeffer, H. Towards a foundation model for partial differential equations: Multi-operator learning and extrapolation, 2025. URL https://arxiv.org/abs/2404.12355.

Takamoto, M., Praditia, T., Leiteritz, R., MacKinlay, D., Alesiani, F., Pflüger, D., and Niepert, M. Pdebench: An extensive benchmark for scientific machine learning, 2024. URL https://arxiv.org/abs/2210.07182.

TransferLab. pyDVL, April 2024. URL https://github.com/aai-institute/pyDVL.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

Wang, S., Bhartari, A. K., Li, B., and Perdikaris, P. Gradient alignment in physics-informed neural networks: A second-order optimization perspective, 2025. URL https://arxiv.org/abs/2502.00604.

Ye, Z., Huang, X., Chen, L., Liu, H., Wang, Z., and Dong, B. PDEformer: Towards a foundation model for one-dimensional partial differential equations. In *ICLR 2024 Workshop on AI4DifferentialEquations In Science*, 2024. URL https://openreview.net/forum?id=GLDMCwdhTK.

Zielinski, P., Krishnan, S., and Chatterjee, S. Weak and strong gradient directions: Explaining memorization, generalization, and hardness of examples at scale, 2020. URL https://arxiv.org/abs/2003.07422.

A. Data Selection.

We emphasize that PDEGym is uniquely suited to the present study because it provides large-scale collections of qualitatively distinct initial condition classes corresponding to evolution of the same governing equations. This structure permits controlled investigation of cross-class gradient coherence and generalization across initial conditions, which is central to our influence-based analysis of solution operator generalization capabilities. While other benchmarks, including PDEBench (Takamoto et al., 2024) and PDEArena (Gupta & Brandstetter, 2022) contain broad and valuable collections of PDE data, PDEGym is presently the only benchmark that simultaneously provides the scale, diversity, and controlled class structure required for the cross-class response analyses developed in this work.

PDEGym data is particularly valuable for studying the progression from a linear wave regime with discontinuities to fully developed turbulence, a crossover that poses computational and analytical challenges due to the presence of sharp wave-fronts and emergent nonlinear interactions. While the CE flows exhibit comparable large-scale structures, they also display qualitatively distinct behaviors. In particular, CE-RPUI initial configurations give rise to complex finger-like instabilities in the flow field that are absent or less pronounced in both CE-RP and CE-CRP. In total, we used $6,500$ trajectories for each of the three classes of initial conditions; for each trajectory, we used the first 16 time steps, for total of approximately $110,000$ training pairs requiring greater than 150 GB memory. We treated the final 4 time-steps as holdout data to test model extrapolation capabilities [see Figure 33, Figure 34, Figure 35, Figure 36].

B. Loss Sensitivity as Gradient Overlap

The influence diagnostic admits a simple interpretation as a first-order sensitivity of the loss function. Let C_p denote the cost associated with a perturbing example p, and let the induced parameter displacement be $\delta\theta_p = -\eta^{-1}\nabla_\theta C_p$, where η is the hat-matrix metric, defined in Equation 4. The corresponding first-order change in the cost of a response example r is $C_r(\theta + \delta\theta_p) - C_r(\theta) \approx -\left\langle \nabla_\theta C_r, \eta^{-1}\nabla_\theta C_p \right\rangle$. Thus, gradient alignment measures the first-order sensitivity of example r to an update generated by example p. Positive alignment indicates that the perturbing update acts coherently on the response example, whereas weak or negative alignment indicates that the corresponding learning signals are decoupled or conflicting.

C. Hat-Matrix

To obtain the related expression familiar from classical influence function theory, let $Q = \hat{y}$ and

$$\delta C = \frac{\delta C}{\delta y}\delta y, \tag{9}$$

where δy is a target feature variation. Then

$$\delta\hat{y}^n = \mathcal{L}_V \hat{y}^n = -\Pi^{nl}\frac{\delta^2 C}{\delta\hat{y}^l \delta y^m}\delta y^m; \tag{10}$$

when C is the mean squared error cost, $\delta\hat{y}/\delta y = \Pi$. Equation 9 allows for investigating the influence of both physics-informed and numerical-routine aware data modifications, which we save for future work.

D. Supplementary Figures

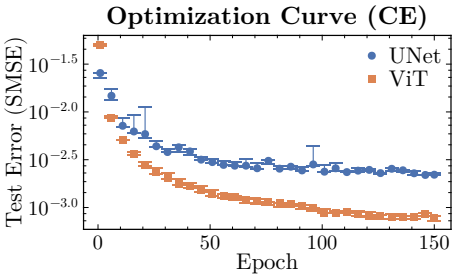

Figure 7. Optimization trajectories for the CE task, shown as test scaled mean squared error (SMSE) versus training epoch for our UNets and our ViTs. Markers denote the median performance, while vertical range bars indicate the variance across model seeds at each epoch. Training is halted in the late-stage learning regime to test whether the induced local gradient geometry is sufficient to identify a physically consistent, generalizing solution.

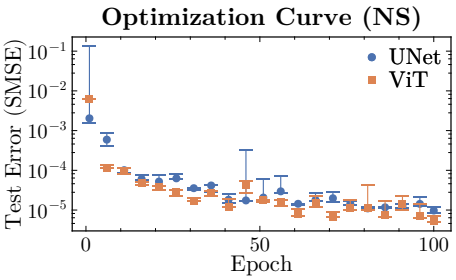

Figure 8. Same as Figure 7, except for NS data. In contrast to the CE problem, which involves four fields and has greater variability in time, one-step NS emulation is learned to comparable proficiency by both models. For representative rollout predictions, see Figure 35 and Figure 36.

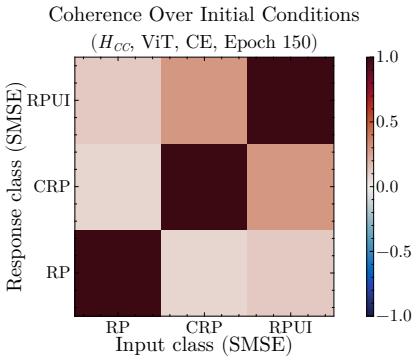

Figure 9. Class-to-class transferability matrix for our ViTs trained on the three-class CE split labeled RP, CRP, and RPUI. Each entry reports the time-averaged response of a given class (vertical axis) produced by test example example gradients from the input class (horizontal axis). Diagonal dominance indicates class-locked gradient geometry; substantial off-diagonal values would imply reuse of dynamical features across classes.

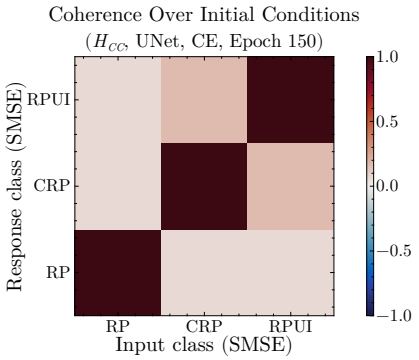

Figure 10. Class-to-class transferability matrix for our UNets trained on the CE split, with the same interpretation as Figure 9.

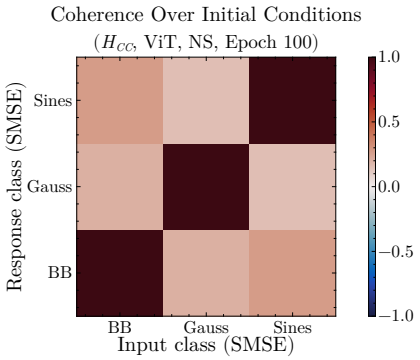

Figure 11. Same as Figure 9 except for NS data. Class-to-class transferability matrix for our ViTs under the alternate three-class NS split labeled BB, Gauss, and Sines. Each cell is a time-averaged influence from one class to another, summarizing whether the learned representation supports feature sharing between qualitatively distinct initial-condition families. Relatively weak off-diagonal structure indicates that training organizes the data into largely disjoint gradient sectors.

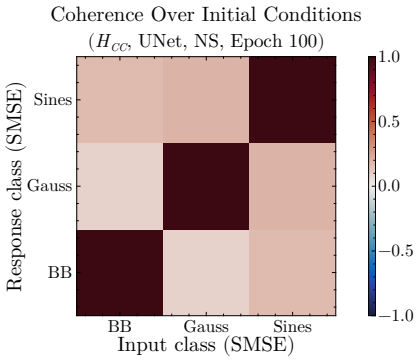

Figure 12. Same as Figure 11 except for out UNets.

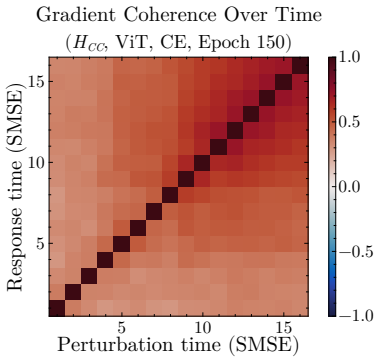

Figure 13. Heatmap of two-time influence for our ViTs trained on CE data, shown as a function of perturbation time (horizontal axis) and response time (vertical axis). Each pixel reports the intra-class averaged response induced by test example gradients at the perturbation time and the resulting change in test loss at the response time. A narrow diagonal ridge corresponds to time-local sensitivity consistent with interpolation, rather than generalization; substantial off-diagonal structure would indicate time-transferable learning.

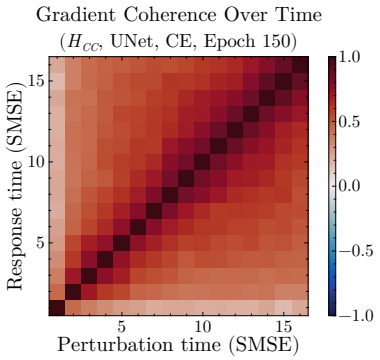

Figure 14. Same as Figure 13, except for our UNets. Desired off-diagonal coherence would indicate genuine temporal generalization.

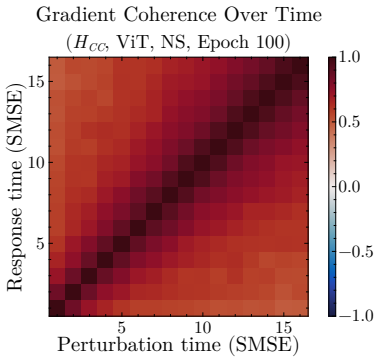

Figure 15. Same as Figure 13, except for NS data. The structure measures how test-time gradient information at one stage of a rollout affects loss at another stage.

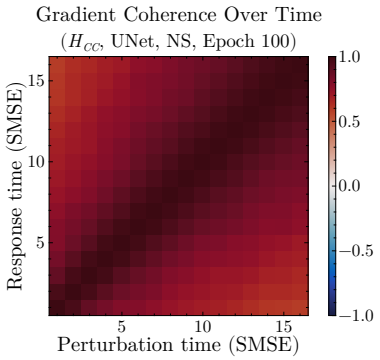

Figure 16. Same as Figure 15, except for our UNets. The enhanced temporal coherence relative to compressible Euler is consistent with viscosity-regularized dynamics.

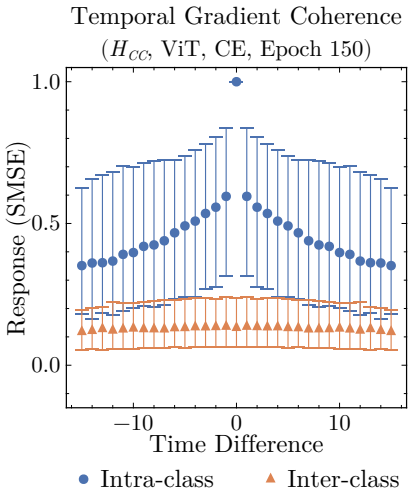

Figure 17. Time-lag summary of temporal transferability for our ViTs on CE data. The influence is averaged over all time-pairs of the same time difference and then split into intra-class pairs (gradient and response drawn from the same initial-condition class) and inter-class pairs (distinct initial-condition classes). Strong concentration near zero time difference indicates that gradient information fails to propagate coherently across time, while the lack off inter-class influence reveals an absence of physics consistent generalization.

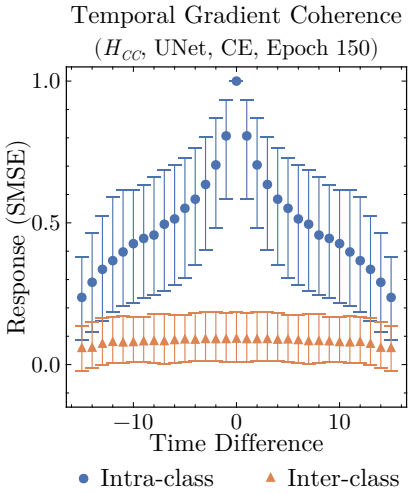

Figure 18. Same as Figure 17, except for our UNets. The gap between curves demonstrates that temporal coherence is predominantly a within-class phenomenon; the model does not learn to share dynamical structure across classes.

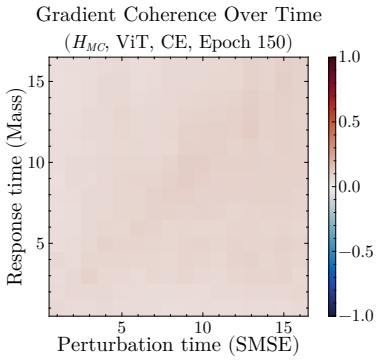

Figure 19. Two-time influence map for our ViTs on CE data at epoch 150 when the response observable is the global mass-consistency signal. The horizontal axis indexes the time at which a test perturbation is applied, and the vertical axis indexes the time at which the mass-based response is evaluated. Off-diagonal support indicates that intra-class mass-related gradient information couples distant times.

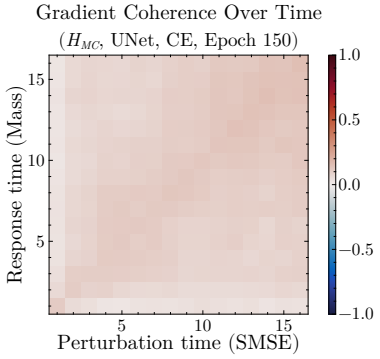

Figure 20. Same as Figure 19, except for our UNets. The response measures the coupling between SMSE perturbations and the mass-consistency observable across time.

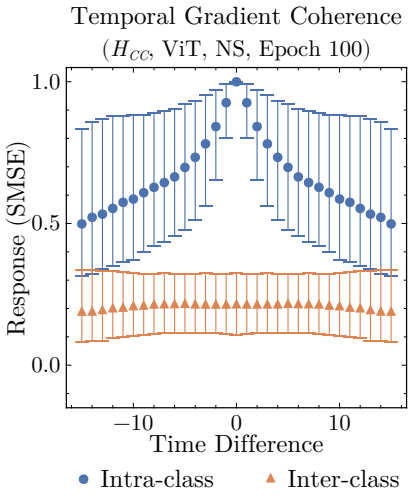

Figure 21. Same as Figure 17, except for NS data. The intra-class and inter-class averages separate temporal coherence within an initial-condition family from cross-family reuse. Persistent suppression of the inter-class curve indicates that the model behaves as a collection of class-conditioned local estimators rather than a shared dynamical emulator.

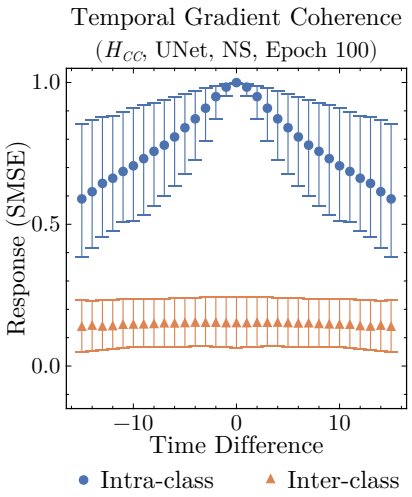

Figure 22. Same as Figure 21, except for our UNets. Inter-class response remains suppressed relative to intra-class response across all time lags.

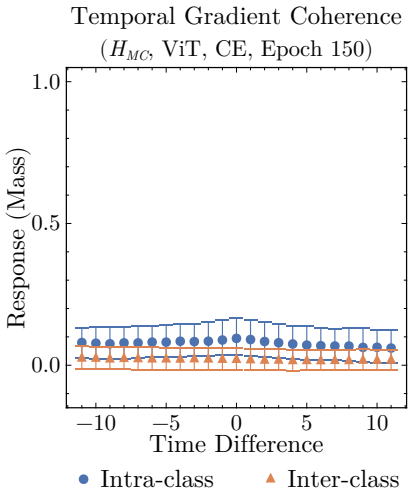

Figure 23. Time-lag influence map for our ViTs on CE data at epoch 150 when the response observable is the global mass-consistency signal. The intra-class response exceeds the inter-class response across all time lags, indicating that coupling between SMSE and mass-consistency gradients is strongest within a given initial-condition class.

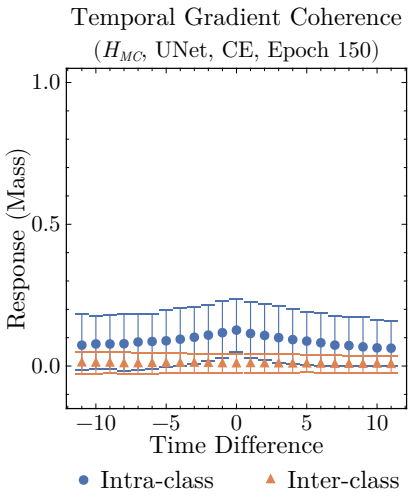

Figure 24. Same as Figure 23, except for our UNets. The intra-class response likewise exceeds the inter-class response across all time lags.

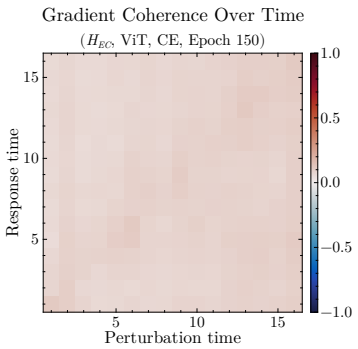

Figure 25. Two-time influence map for our ViTs on CE data at epoch 150 when the response observable is the global energy-consistency signal. The horizontal axis indexes the time at which a test perturbation is applied, and the vertical axis indexes the time at which the energy-based response is evaluated. Off-diagonal support indicates that intra-class energy-related gradient information couples distant times.

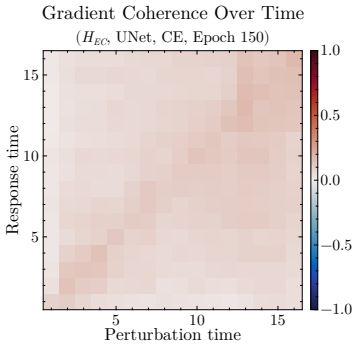

Figure 26. Same as Figure 25, except for our UNets. The response measures the coupling between SMSE perturbations and the energy-consistency observable across time.

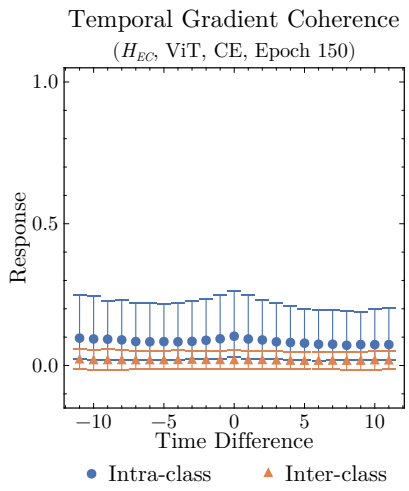

Figure 27. Time-lag influence map for our ViTs on CE data at epoch 150 when the response observable is the global energy-consistency signal. The separation between intra-class and inter-class response indicates that cross-objective gradient coupling remains predominantly within a given initial-condition class.

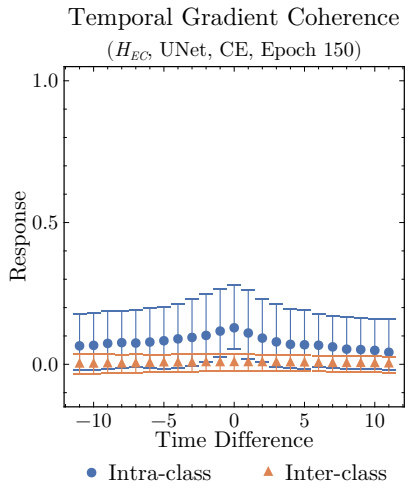

Figure 28. Same as Figure 27, except for our UNets. The separation between intra-class and inter-class response is also observed for the UNet energy-consistency observable.

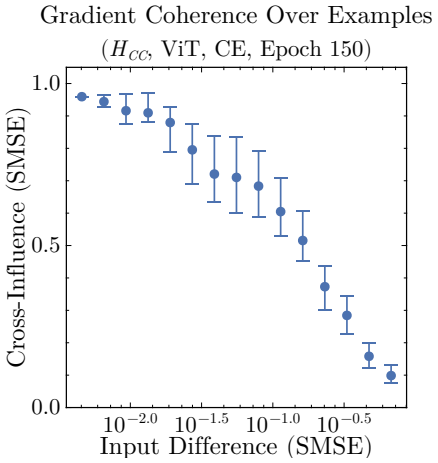

Figure 29. Influence as a function of feature-space separation between input states for CE data for our ViT; rangebars show variability across seeds. Decay to zero indicates short-range locality on the learned data manifold, implying that parameter updates affect only nearby states and generalization is limited. For NS counterpark, see Figure 31.

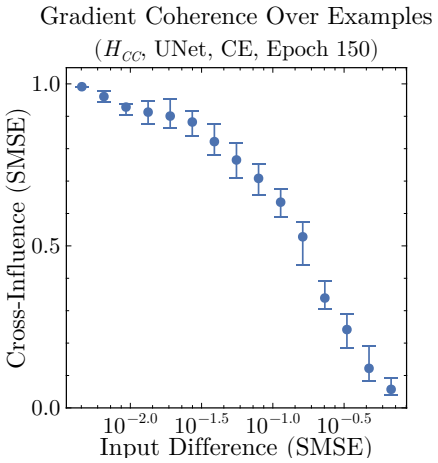

Figure 30. Same as Figure 29, except for our UNets. The influence exhibits a similar decay with feature-space separation, indicating that locality of the response geometry is not specific to the ViT architecture.

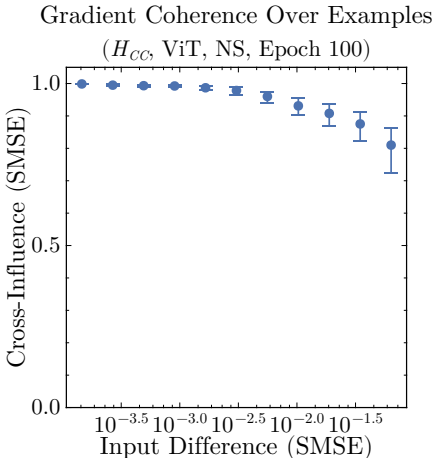

Figure 31. Influence as a function of feature-space separation between input states for NS data with our ViT; range bars show variability across seeds. In contrast to the CE case, the response remains nearly constant over the available range of NS separations. This weaker distance dependence is consistent with the smaller spread of NS states in feature space, whose maximum separation is approximately an order of magnitude below that observed for CE flows.

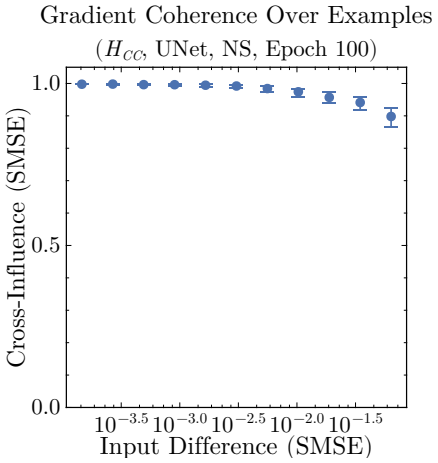

Figure 32. Same as Figure 30, except for NS data. As with the ViT, the response exhibits only weak dependence on feature-space separation over the range probed by the NS trajectories.

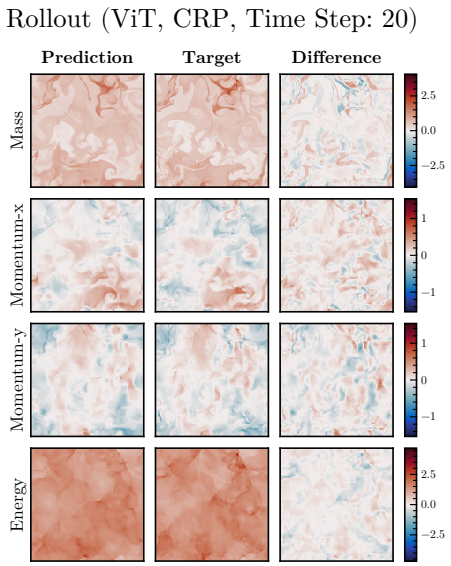

Figure 33. Representative final time rollout prediction for one of our ViT models on a CE-CRP initial condition.

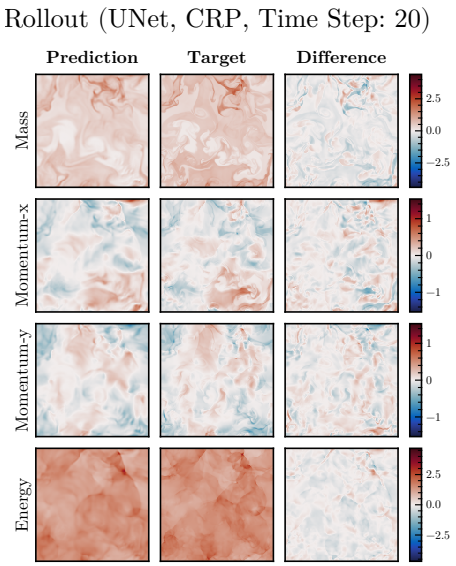

Figure 34. Same as Figure 33 except for one of our UNets. Relative to the ViT prediction, the UNet renders less sharply resolved flow features at the final rollout time.

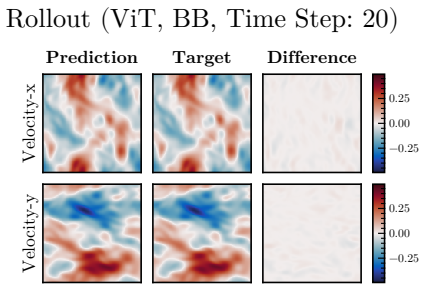

Figure 35. Representative final time rollout prediction for one of our ViT models on a NS-BB initial condition.

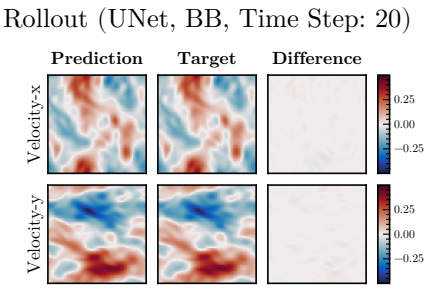

Figure 36. Same as Figure 35 except for one of our UNets. The prediction remains visually consistent with the target flow field at the final rollout time.

Table 2. Correlation between cross-time influence and cumulative rollout error as a function of extrapolation time on CE trained models. For NS data companion figure, see Table 3.

Rollout Time	1	2	3	4	5	6	7	8	9	10
ViT	-0.67	-0.67	-0.67	-0.66	-0.65	-0.64	-0.63	-0.63	-0.62	-0.62
UNet	-0.59	-0.57	-0.53	-0.50	-0.48	-0.45	-0.42	-0.40	-0.37	-0.35

Table 3. Correlation between cross-time influence and rollout error as a function of extrapolation time for NS models.

Rollout Time	1	2	3	4	5	6	7	8	9	10
ViT [NS]	-0.69	-0.63	-0.57	-0.53	-0.50	-0.48	-0.47	-0.46	-0.47	-0.47
UNet [NS]	-0.50	-0.47	-0.44	-0.42	-0.41	-0.40	-0.38	-0.36	-0.36	-0.37

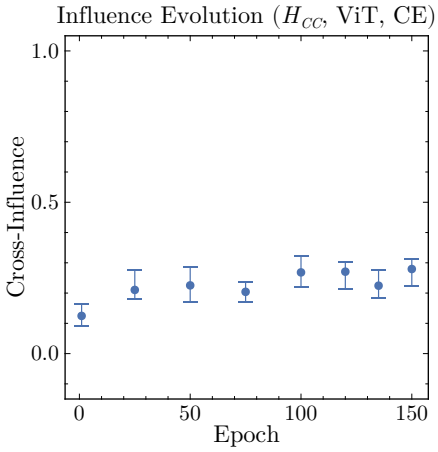

Figure 37. Off-diagonal components of the two-time response across checkpointed training epochs. The off-diagonal response remains largely constant throughout optimization, indicating that cross-time gradient coupling is established early and does not substantially reorganize during training.

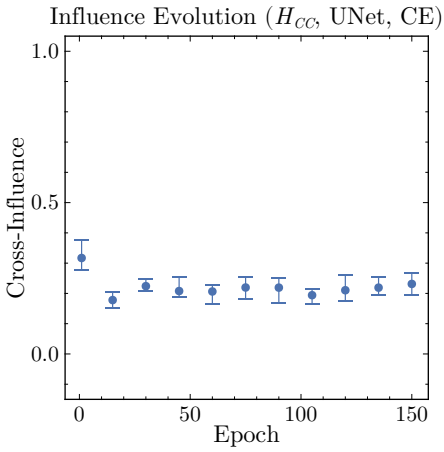

Figure 38. Same as Figure 37 except for our UNets. The suppressed off-diagonal influence persists throughout training, indicating that the localized response geometry is not specific to the ViT architecture.

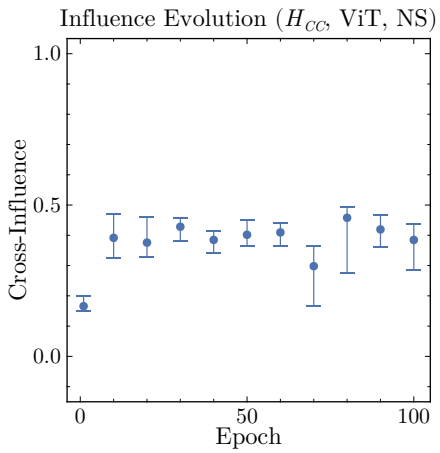

Figure 39. Off-diagonal components of the two-time response across checkpointed training epochs for the NS ViT. In contrast to the CE case, the off-diagonal response remains largely constant at values near 0.4, consistent with smoother viscous dynamics and reduced variation between adjacent time steps.

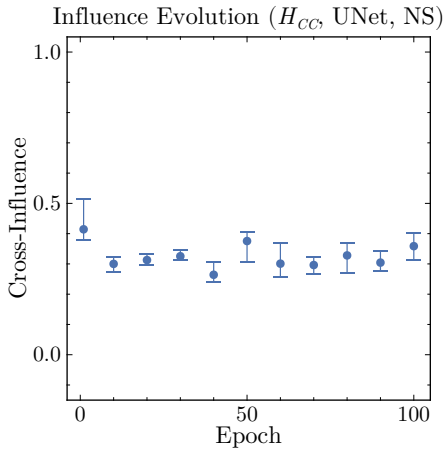

Figure 40. Same as Figure 39 except for our UNets. As with the ViT, the off-diagonal influence contributions remain comparatively constant in magnitude throughout training.

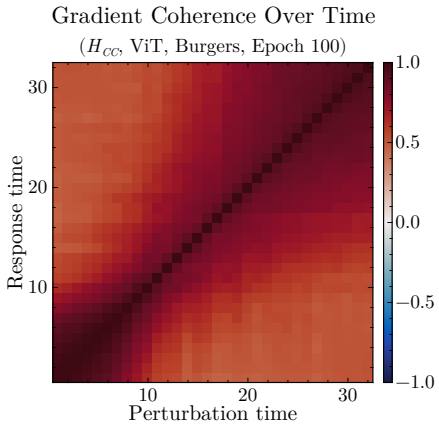

Figure 41. Heatmap of two-time influence for our ViTs trained on Burgers data, shown as a function of perturbation time (horizontal axis) and response time (vertical axis). Extended off-diagonal support is consistent with our findings that this model is capable of capable of generating accurate long-horizon extrapolations in time.

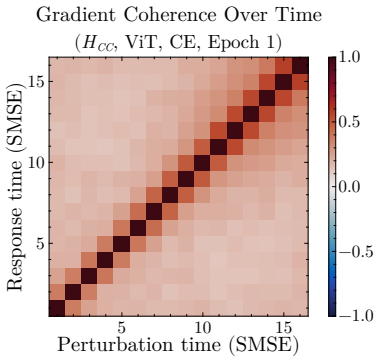

Figure 42. Heatmap of two-time influence (H_{CC}) for our ViTs trained on CE data, shown as a function of perturbation time (horizontal axis) and response time (vertical axis). Each pixel reports the intra-class averaged response induced by test example gradients at the perturbation time. A narrow diagonal ridge corresponds to time-local sensitivity consistent with interpolation, rather than generalization; substantial off-diagonal structure would indicate time-transferable learning.

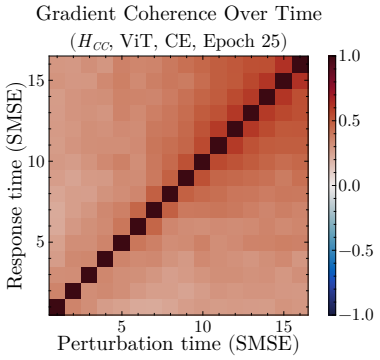

Figure 43. Heatmap of two-time influence (H_{CC}) for our ViTs trained on CE data, shown as a function of perturbation time (horizontal axis) and response time (vertical axis). Each pixel reports the intra-class averaged response induced by test example gradients at the perturbation time. A narrow diagonal ridge corresponds to time-local sensitivity consistent with interpolation, rather than generalization; substantial off-diagonal structure would indicate time-transferable learning.

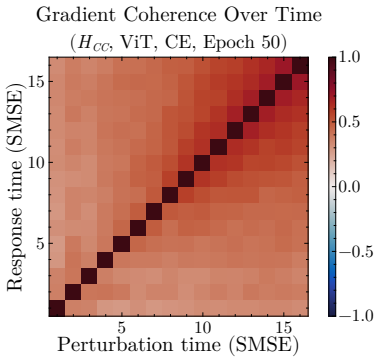

Figure 44. Heatmap of two-time influence (H_{CC}) for our ViTs trained on CE data, shown as a function of perturbation time (horizontal axis) and response time (vertical axis). Each pixel reports the intra-class averaged response induced by test example gradients at the perturbation time. A narrow diagonal ridge corresponds to time-local sensitivity consistent with interpolation, rather than generalization; substantial off-diagonal structure would indicate time-transferable learning.

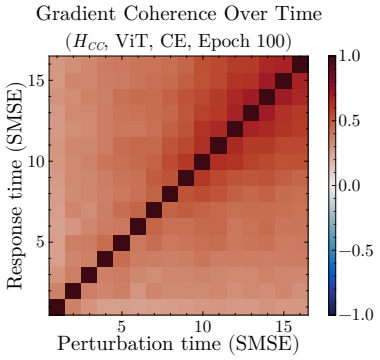

Figure 45. Heatmap of two-time influence (H_{CC}) for our ViTs trained on CE data, shown as a function of perturbation time (horizontal axis) and response time (vertical axis). Each pixel reports the intra-class averaged response induced by test example gradients at the perturbation time. A narrow diagonal ridge corresponds to time-local sensitivity consistent with interpolation, rather than generalization; substantial off-diagonal structure would indicate time-transferable learning.

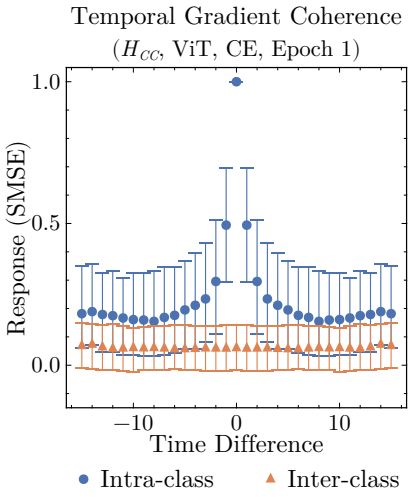

Figure 46. Time-lag summary of temporal transferability for our ViTs on CE data. The influence (H_{CC}) is averaged over all time-pairs of the same time difference and then split into intra-class pairs (gradient and response drawn from the same initial-condition class) and inter-class pairs (distinct initial-condition classes). Strong concentration near zero time difference indicates that gradient information fails to propagate coherently across time, while the lack off inter-class influence reveals an absence of physics consistent generalization.

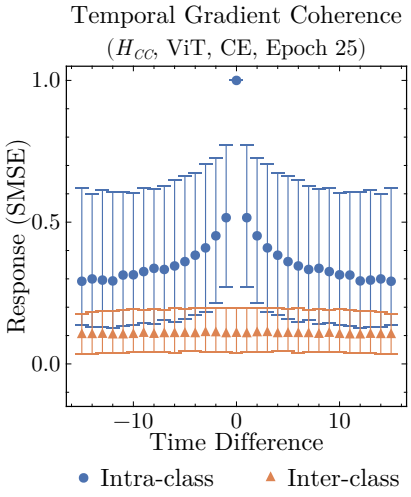

Figure 47. Time-lag summary of temporal transferability for our ViTs on CE data. The influence (H_{CC}) is averaged over all time-pairs of the same time difference and then split into intra-class pairs (gradient and response drawn from the same initial-condition class) and inter-class pairs (distinct initial-condition classes). Strong concentration near zero time difference indicates that gradient information fails to propagate coherently across time, while the lack off inter-class influence reveals an absence of physics consistent generalization.

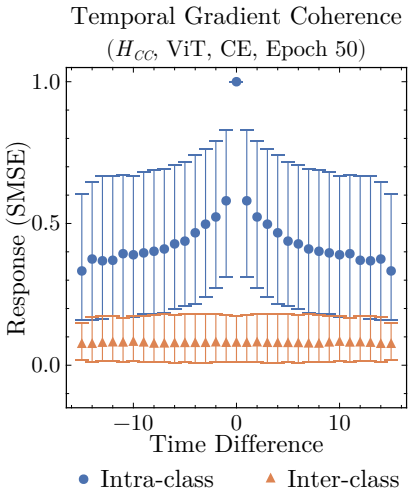

Figure 48. Time-lag summary of temporal transferability for our ViTs on CE data. The influence (H_{CC}) is averaged over all time-pairs of the same time difference and then split into intra-class pairs (gradient and response drawn from the same initial-condition class) and inter-class pairs (distinct initial-condition classes). Strong concentration near zero time difference indicates that gradient information fails to propagate coherently across time, while the lack off inter-class influence reveals an absence of physics consistent generalization.

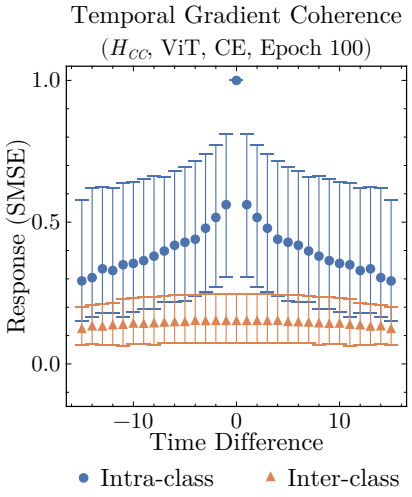

Figure 49. Time-lag summary of temporal transferability for our ViTs on CE data. The influence (H_{CC}) is averaged over all time-pairs of the same time difference and then split into intra-class pairs (gradient and response drawn from the same initial-condition class) and inter-class pairs (distinct initial-condition classes). Strong concentration near zero time difference indicates that gradient information fails to propagate coherently across time, while the lack off inter-class influence reveals an absence of physics consistent generalization.

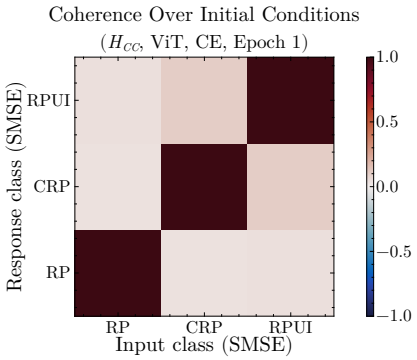

Figure 50. Class-to-class transferability matrix for our ViTs trained on the three-class CE split labeled RP, CRP, and RPUI. Each entry reports the time-averaged influence (H_{CC}) of test examples from the input class (horizontal axis) on examples from the response class (vertical axis). Diagonal dominance indicates class-locked gradient geometry; substantial off-diagonal values would imply shared learning across classes.

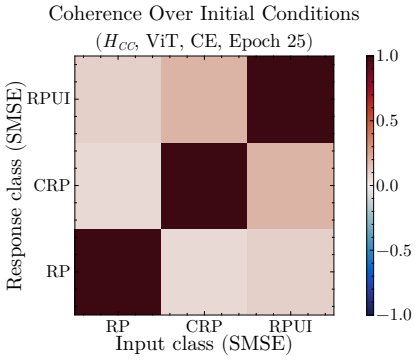

Figure 51. Class-to-class transferability matrix for our ViTs trained on the three-class CE split labeled RP, CRP, and RPUI. Each entry reports the time-averaged influence (H_{CC}) of test examples from the input class (horizontal axis) on examples from the response class (vertical axis). Diagonal dominance indicates class-locked gradient geometry; substantial off-diagonal values would imply shared learning across classes.

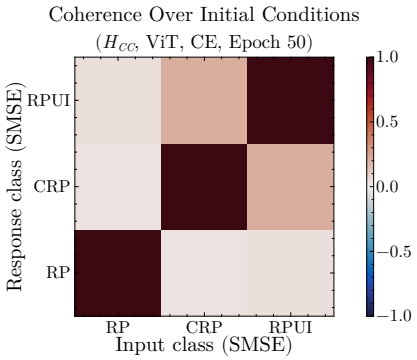

Figure 52. Class-to-class transferability matrix for our ViTs trained on the three-class CE split labeled RP, CRP, and RPUI. Each entry reports the time-averaged influence (H_{CC}) of test examples from the input class (horizontal axis) on examples from the response class (vertical axis). Diagonal dominance indicates class-locked gradient geometry; substantial off-diagonal values would imply shared learning across classes.

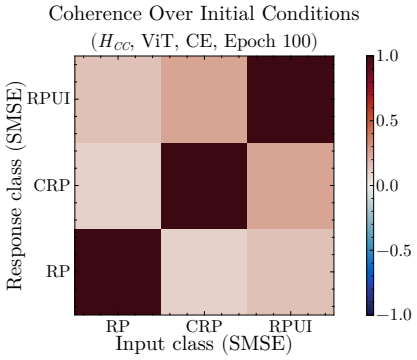

Figure 53. Class-to-class transferability matrix for our ViTs trained on the three-class CE split labeled RP, CRP, and RPUI. Each entry reports the time-averaged influence (H_{CC}) of test examples from the input class (horizontal axis) on examples from the response class (vertical axis). Diagonal dominance indicates class-locked gradient geometry; substantial off-diagonal values would imply shared learning across classes.

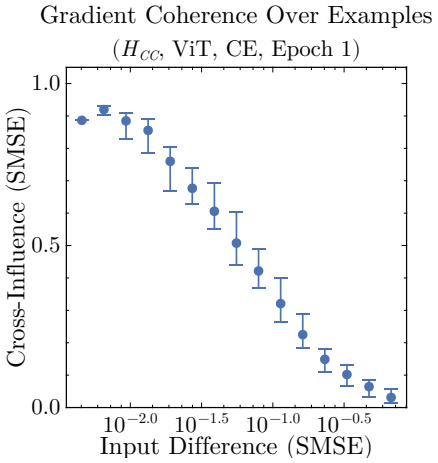

Figure 54. Influence as a function of feature-space separation between input states for CE data, comparing UNet and ViT at; rangebars show variability across seeds. A steep decay indicates short-range interpolation of influence across the learned data manifold, implying that parameter updates affect only nearby states and generalization is limited.

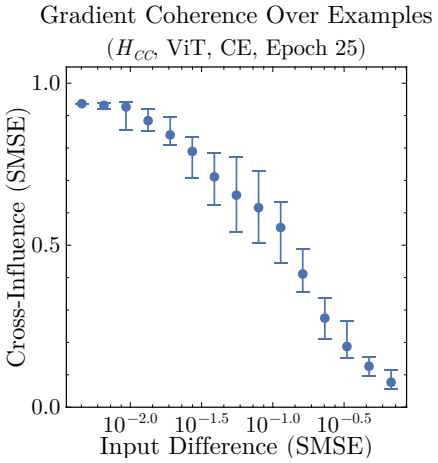

Figure 55. Influence as a function of feature-space separation between input states for CE data, comparing UNet and ViT at; rangebars show variability across seeds. A steep decay indicates short-range interpolation of influence across the learned data manifold, implying that parameter updates affect only nearby states and generalization is limited.

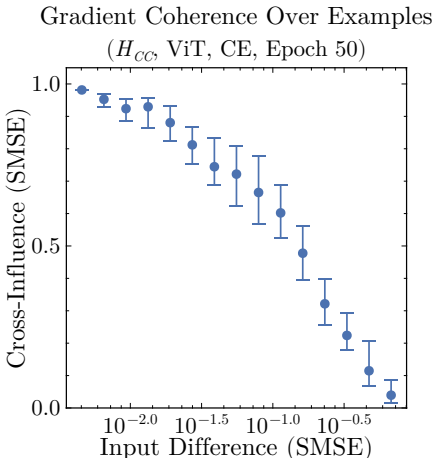

Figure 56. Influence as a function of feature-space separation between input states for CE data, comparing UNet and ViT at; rangebars show variability across seeds. A steep decay indicates short-range interpolation of influence across the learned data manifold, implying that parameter updates affect only nearby states and generalization is limited.

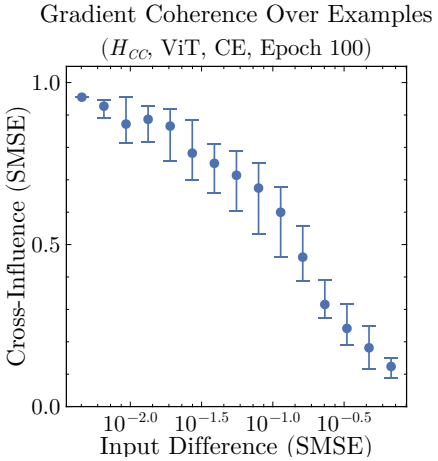

Figure 57. Influence as a function of feature-space separation between input states for CE data, comparing UNet and ViT at; rangebars show variability across seeds. A steep decay indicates short-range interpolation of influence across the learned data manifold, implying that parameter updates affect only nearby states and generalization is limited.