

Skeptical review: Mathematical Interpretation of PINN Latent Space for Burger’s Equation: Learned Dynamics and Geometric Structure

Summary

The manuscript proposes a quantitative framework to interpret the internal 10D latent representation $L(x, t)$ of a Physics-Informed Neural Network (PINN) trained on the 1D viscous Burgers’ equation on a 2D (x, t) grid (Secs. 1–2.1). From a provided $100 \times 100 \times 10$ latent dataset, it computes numerical derivatives $V_x = \partial L / \partial x$, $V_t = \partial L / \partial t$, and $V_{xx} = \partial^2 L / \partial x^2$ via finite differences (Sec. 2.3), then analyzes latent statistics and geometry (correlations, norms, cosine similarities, and local tangent-space SVD of $[V_x | V_t]$) (Secs. 2.2–2.5, 3.1–3.3). Finally, it performs a SINDy-like sparse regression (Lasso) with a hand-designed candidate library in L , V_x , and V_{xx} to infer latent PDEs of the form $\partial L_k / \partial t = f_k(L, V_x, V_{xx})$, reporting $R^2 \approx 0.81$ – 0.93 and advection-/diffusion-like terms reminiscent of Burgers’ nonlinearity and viscosity (Secs. 2.6, 3.4). The overall direction—combining geometric diagnostics with sparse latent dynamics discovery for PINN interpretability—is timely and promising, but the current version is difficult to assess rigorously because the PINN/latent definition is under-specified, derivative accuracy is not validated (despite PINNs permitting autodiff), the sparse-regression library and evaluation protocol are internally inconsistent/underspecified, and several key interpretability claims (e.g., “encoding physical principles,” near-1D tangent spaces) would benefit from stronger controls, invariance considerations, and robustness studies (Secs. 3.3–3.5, 4).

Strengths

- Targets an important ‘mechanistic interpretability’ question for PINNs by analyzing learned internal representations rather than only the output solution (Sec. 1).
- Clear end-to-end pipeline: latent extraction \rightarrow numerical derivatives \rightarrow descriptive + geometric analyses \rightarrow sparse latent PDE discovery (Sec. 2; Secs. 3.1–3.4).
- Uses Burgers’ equation as a well-motivated testbed where advection/diffusion analogies in discovered terms are intuitive and potentially meaningful (Sec. 1; Sec. 3.4).
- Reports multiple complementary quantitative diagnostics (norms, cosine similarities, singular-value ratios, R^2 , sparsity) that help assess the presence/strength of structure in the latent field (Secs. 3.1–3.4).
- Mathematical objects (L , V_x , V_t , V_{xx}) and core formulas for norms/cosines and the regression objective are largely correctly stated (Secs. 2.3–2.6).
- Figures (esp. the multi-figure sequence 1–10) attempt to make latent-space statistics and geometry visually accessible and, with polishing, can be a strong asset for readers.

Major issues

1. **PINN architecture, training setup, and the precise definition of the latent representation $L(x, t)$ are not specified, making the latent analyses hard to interpret and impossible to reproduce (Sec. 1, Sec. 2.1).** It is unclear which layer defines L (before/after nonlinearity), whether L is a bottleneck, and how $u(x, t)$ is computed from L (linear readout vs further nonlinear layers). This matters because latent coordinates are not unique: invertible linear transforms (rotations/scalings) can dramatically change correlations and the apparent presence/shape of terms like $L_j V_{x,j}$, undermining ‘physical structure’ claims unless invariances are addressed.

Recommendation: Expand Sec. 2.1 into a complete model/training description: layer-by-layer architecture (width/depth, activations), the exact layer and operation defining $L(x, t)$, and the mapping from L to $u(x, t)$. Specify loss terms and weights (PDE residual vs data/BC/IC), collocation/data sampling strategy, optimizer/schedule, stopping criteria, random seeds. Add an explicit discussion of latent non-identifiability (Sec. 4): which conclusions are basis-dependent, and add at least one ‘gauge-fixing’ or invariance check (e.g., PCA-whiten L before analysis; or apply random orthogonal transforms to L and show which geometric/regression conclusions persist).

2. **The work does not quantitatively establish that the underlying PINN solution $u(x, t)$ is accurate for the stated Burgers setup, nor does it specify the PDE parameters and conditions (ν , domain bounds, IC/BC) that strongly shape gradients/shock behavior (Sec. 1, Sec. 2.1, Sec. 3).** Without this, it is unclear whether observed latent structures reflect the intended PDE solution or training artifacts/underfitting.

Recommendation: In Sec. 2.1 (or a new ‘Problem setup’ subsection), state ν , x/t ranges, and all IC/BC used. In Sec. 3 (before Sec. 3.1), report $u(x, t)$ accuracy vs an analytic or high-resolution numerical reference on the same grid (relative L_2 , L_∞ , and possibly error heatmaps). If the latent dataset is from a ‘pre-trained’ model, state provenance clearly (trained by authors vs external), and provide enough details (or a release link) to reproduce it.

3. **Finite-difference derivatives (especially V_{xx}) are central to both geometry and sparse regression, but the derivative computation is insufficiently specified and not validated; boundary handling is unclear; and noise amplification may strongly affect reported large V_{xx} magnitudes and discovered latent PDE terms (Sec. 2.3, Sec. 3.2, Sec. 3.4).** Given PINNs permit automatic differentiation, relying only on finite differences without cross-checks weakens credibility.

Recommendation: Augment Sec. 2.3 with explicit Δx , Δt , uniformity assumptions, interior stencils, and boundary stencils or masking rules (for V_x , V_t , V_{xx}). Then add a validation study: (i) compare FD derivatives to autodiff derivatives from the PINN at

a subset of grid points (or on the full grid if feasible); (ii) rerun key statistics and the regression with alternative stencils/smoothing and report sensitivity (Sec. 3.2, Sec. 3.4). Clearly state whether boundary points are excluded from Θ and metrics; if masked, update array-shape descriptions accordingly.

4. **The tangent-space ‘near-1D’ conclusion from SVD of $M = [V_x|V_t]$ (Secs. 2.5, 3.3.3) is potentially confounded by (a) rescaling of t vs x (σ_2/σ_1 is not invariant to $t \rightarrow ct$), (b) regions where $\|V_x\|$ and $\|V_t\|$ are near zero (ratio unstable), and (c) the possibility that $V_t \approx -c(x,t)V_x$ simply reflects advective transport (a generic property of traveling/shock-like structures) rather than a special latent-manifold property.**

Recommendation: In Sec. 2.5, state the scaling/nondimensionalization of x and t (or report a sensitivity analysis of σ_2/σ_1 under rescaling). In Sec. 3.3.3, exclude or separately analyze points where $\| [V_x, V_t] \|$ is below a threshold to avoid ratio blow-ups, and provide (x, t) heatmaps of σ_2/σ_1 with this masking. Strengthen interpretation by: (i) computing the analogous analysis for the output $u(x, t)$ (use $[u_x, u_t]$) to test whether the effect is latent-specific; (ii) if $V_t \approx -cV_x$, estimate $c(x, t)$ (local wave speed) and relate it to known Burgers dynamics (Sec. 3.5).

5. **Sparse-regression (SINDy/Lasso) methodology is under-specified and internally inconsistent regarding the candidate library Θ and term counts (Sec. 2.6.2 vs Sec. 3.4), and lacks essential details for identifiability and generalization: feature scaling/standardization (critical with large V_{xx} ranges), α selection, solver settings, train/validation splitting, and stability under multicollinearity (Secs. 2.6.2–2.6.3, 3.4).** As written, the reported R^2 may be in-sample and inflated by spatiotemporal autocorrelation.

Recommendation: Make Θ explicit and consistent: list all included term families and whether cross-channel terms ($j \neq m$) are included, and show the resulting column count (e.g., reconcile the ‘61 terms’ claim). In Sec. 2.6.3, specify preprocessing (standardize Θ columns and targets; intercept handling), Lasso implementation details (package/solver/max_iter/tol), and a principled α selection (cross-validation, information criteria, or stability selection). In Sec. 3.4, use a documented train/test protocol (e.g., hold out time slices or spatial blocks to reduce leakage), and report R^2 /RMSE on both. Add coefficient/term stability via bootstrapping or comparing Lasso vs Elastic Net / sequential thresholded least squares (classic SINDy).

6. **Claims that the discovered latent PDEs ‘encode key physical principles’ are currently qualitative and may be partly driven by library design and latent coordinate choices (Secs. 3.4–3.5, 4).** There is no systematic comparison to the true Burgers operator, no quantification of term importance across latent dimensions, and no demonstration that the discovered latent PDEs can be integrated to reproduce latent trajectories or reconstruct $u(x, t)$.

Recommendation: Strengthen Sec. 3.4–3.5 with quantitative, falsifiable tests: (i) provide per-latent-dimension tables of nonzero terms with standardized coefficients and/or variance contributions; (ii) compare the signs/magnitudes of discovered ‘advection’ and ‘diffusion’ terms to uu_x and νu_{xx} computed from u on the same grid; (iii) perform ablations: restrict the library to Burgers-like terms and compare fit; remove V_{xx} or $L_j V_{x,j}$ and quantify degradation; (iv) if feasible, integrate the discovered latent PDE forward in time from initial $L(x, 0)$, decode to u via the PINN, and compare to the PINN/reference solution. If integration is not feasible, temper claims accordingly and frame results as descriptive/operator-similarity evidence rather than recovered physics.

- 7. Generality and controls are missing: all conclusions rely on a single trained PINN / single parameter configuration, with no robustness across seeds/architectures/viscosities and no control comparison to a non-physics-trained network.** Thus it is unclear which observed properties are generic to PINNs, generic to Burgers solutions, or idiosyncratic to this run (Secs. 3.1–3.5, 4).

Recommendation: Add a minimal robustness/control suite (Sec. 3.5 or Appendix): re-train with 2–5 seeds and report variation in key metrics (correlations, $|V_x|$ vs $|V_t|$, $\cos(V_x, V_t)$, σ_2/σ_1 , and dominant regression terms). If feasible, vary ν (or compare to another PDE) and show the discovered term structure changes appropriately. Add a control model: same architecture trained supervised-only on u data (no PDE residual) and compare whether ‘physical’ latent PDE structure persists—this directly supports (or challenges) the interpretability thesis.

Minor issues

1. Terminology repeatedly refers to “2D Burgers’ equation,” but the equation is the standard 1D-in-space viscous Burgers equation on a 2D (x, t) domain (Abstract, Sec. 1, Eq. (1)). This can mislead readers into thinking of two spatial dimensions.

Recommendation: Revise title/Abstract/Sec. 1 to consistently say “1D viscous Burgers’ equation (in x , with time t)” and, if needed, clarify that “2D” refers only to the (x, t) domain.

2. Global correlation/statistics computed over a spatiotemporally autocorrelated grid can inflate significance and obscure heterogeneity (Sec. 2.2, Sec. 3.1; Fig. 1). Also, correlation heatmaps are basis-dependent under latent rotations.

Recommendation: Report effective sample size or uncertainty via block bootstrap / thinning. Complement Pearson correlations with basis-invariant analyses (PCA spectrum, explained variance) and clearly state dependence on latent basis in Fig. 1 caption. Consider stratifying statistics by time slices or shock vs smooth regions.

3. Spatial heterogeneity is not well localized: many reported effects (anti-alignment, large V_{xx} , near-1D tangent) are presented via histograms without clear (\mathbf{x}, t) maps, making it hard to relate observations to Burgers solution features (Secs. 3.2–3.3; Figs. 5–10).

Recommendation: Add (\mathbf{x}, t) heatmaps for representative quantities ($|V_x|$, $|V_t|$, $|V_{xx}|$, $\cos(V_x, V_t)$, σ_2/σ_1 , and regression residual magnitude) to connect findings to shock location/boundaries (Sec. 3).

4. Handling of boundary points for V_{xx} and inclusion/exclusion in later analyses remains vague (Sec. 2.3.3, Secs. 3.2–3.4).

Recommendation: State explicitly whether boundaries are masked, use one-sided stencils, or are excluded from Θ /regression and from which figures/metrics. Ensure reported shapes/flattening reflect the actual used set of points.

5. Dimensionality claims rely mainly on correlations and local σ_2/σ_1 ; a global intrinsic-dimensionality estimate is missing (Sec. 3.1, Sec. 3.3.3, Sec. 3.5).

Recommendation: Add PCA on flattened $L(\mathbf{x}, t)$ vectors and report cumulative explained variance; optionally add an intrinsic-dimension estimator. Relate global results to the local tangent SVD.

6. Regression evaluation likely suffers from leakage due to nearby (\mathbf{x}, t) points being highly correlated; simple random splits can overestimate generalization (Sec. 3.4).

Recommendation: Use blocked splits (hold out contiguous time intervals or spatial bands) and report performance; optionally compare to random splits to illustrate the difference.

7. Related-work section appears unbalanced toward general PCA/latent-space citations (including astronomy/cosmology examples) while key PINN/SINDy/PDE-discovery/latent-dynamics literature is comparatively light (Secs. 1–2).

Recommendation: Add and discuss foundational and closely related references on PINNs, SINDy/PDE-FIND, derivative estimation for system identification, and prior work on interpreting latent spaces of physics-informed/operator-learning models; clarify novelty relative to existing system-ID approaches.

8. Figures 2–4 and others have inconsistent notation (L_0 vs L_0), small fonts, and overplotting that obscures density; captions often omit N , binning, and preprocessing (Figs. 2–4; also general across Figs. 1–10).

Recommendation: Standardize notation across all figures/captions; increase font sizes; use transparency/hexbin for scatter plots; annotate key statistics (Pearson r , N). Add caption details on sampling, binning, and any clipping/normalization.

9. Local tangent-space statistics may be dominated by near-zero-gradient regions, where σ_2/σ_1 is numerically unstable (Sec. 3.3.3).

Recommendation: Report conditional statistics restricted to $\|[\mathbf{V}_x, \mathbf{V}_t]\|$ above a threshold, and show sensitivity to threshold choice.

10. Normalization/scaling of $\mathbf{x}, t, \mathbf{L}$ (and thus derivative magnitudes and regression coefficient magnitudes) is not clearly documented (Secs. 2.1–2.3, 3.3–3.4).

Recommendation: State whether \mathbf{x}, t were mapped to $[0, 1]$ (or nondimensionalized) and whether \mathbf{L} was standardized; describe how this impacts $\mathbf{V}_x, \mathbf{V}_t, \mathbf{V}_{xx}$ and coefficient interpretation.

11. Implementation/reproducibility details (versions, runtime, seeds) are incomplete (Sec. 2.7).

Recommendation: Add Python/library versions, seed handling, approximate runtimes for derivative/SVD/regression steps, and a code/data release plan (or a clear statement if not releasing).

Very minor issues

1. Typographical and consistency issues: “Burger’s/Burgers’”, “PIN/PINN”, inconsistent math formatting for array names, and occasional LaTeX/HTML artifacts (e.g., “> 0.8”) (Secs. 1–4; figure captions).

Recommendation: Proofread to standardize terminology (“Burgers’ equation”), notation, and LaTeX rendering; ensure consistent latent indexing ($\mathbf{k} = 0, \dots, 9$) across text and figures.

2. Section heading formatting inconsistencies and minor bibliography/citation duplication artifacts (Sec. 3 headings; References).

Recommendation: Standardize headings to venue style and clean bibliography entries and year-suffix duplication.

3. A noted reshape/consistency check did not complete due to an execution error (TypeError) (mentioned around Sec. 2.6.1 checks).

Recommendation: Fix and rerun the check; ensure flattening/reshaping conventions used to construct Θ and targets are validated and documented.

4. Component-wise product notation varies ($L_j \mathbf{V}_{x,j}$ vs $L_j \mathbf{V}_{xj}$ vs $L_j(\partial L_j / \partial \mathbf{x})$), which may confuse whether multiplication is componentwise or implies summation (Abstract; Sec. 3.4).

Recommendation: Define a single convention (e.g., $\mathbf{V}_{x,k} := \partial L_k / \partial \mathbf{x}$ and use $L_k \cdot \mathbf{V}_{x,k}$ for componentwise products) and apply it consistently.

Key statements and references

- **△ Physics-Informed Neural Networks (PINNs) embed the governing physical laws, expressed as PDE residuals, directly into the neural network’s loss function alongside data constraints, enabling them to learn solutions to forward problems, discover unknown parameters in inverse problems, and approximate operators using only sparse measurements of the system state, but their internal representations remain largely opaque as highlighted in recent work on PINN interpretability and training strategies.**
 - *Reference(s)*: Dahlbüdding et al., 2024, Hao et al., 2025, Clemente et al., 2025
 - *Justification*: Supported: (i) PINNs embed PDE residuals and boundary/data terms in the loss and are used to solve forward problems (Dahlbüdding et al., 2024, Sec. 2.1; Hao et al., 2025, Sec. 2), (ii) can be used for inverse parameter discovery (Dahlbüdding et al., 2024, Sec. 2.1), (iii) are trained with strategies addressing stability (Hao et al., 2025), and (iv) internal representations are largely opaque (Dahlbüdding et al., 2024, Sec. 4), with broader interpretability motivation in physics-informed ML (Clemente et al., 2025). Not fully supported: the claim that PINNs approximate operators from sparse measurements is not demonstrated or stated for PINNs in these papers (operator learning is only cited generally, not shown for PINNs). Hence, partial support.
- **✘ The 2D Burger’s equation used in this study, $\partial u/\partial t + u \partial u/\partial x - \nu \partial^2 u/\partial x^2 = 0$, is adopted as a testbed because of its well-known balance of nonlinear advection and diffusion and its ability to exhibit wave steepening and shock formation, as discussed in prior analyses of Navier–Stokes-type systems.**
 - *Reference(s)*: McDuffee, 2023, Chiodaroli and Feireisl, 2024
 - *Justification*: Neither McDuffee, 2023 nor Chiodaroli and Feireisl, 2024 mention or use the 2D Burgers equation, nor do they discuss adopting it as a testbed or its properties (nonlinear advection–diffusion balance, wave steepening, shock formation). McDuffee, 2023 focuses on incompressible Navier–Stokes/Einstein duality and gravitational waves, while Chiodaroli and Feireisl, 2024 analyzes long-time behavior of the Navier–Stokes–Fourier system on unbounded domains. Thus the statement is not supported by the attached papers.
- **✘ The exploratory data analysis methodology for the latent variables follows established practices in astronomical and physical data analysis by computing standard descriptive statistics (mean, variance, min, max) and Pearson correlation matrices over high-dimensional latent representations to reveal heterogeneous distributions and strong interdependencies among components.**

- *Reference(s)*: Andrianomena and Hassan, 2023, Iwasaki et al., 2023, Skowronski et al., 2025
- *Justification*: None of the papers describe EDA of latent variables via computing mean/variance/min/max and Pearson correlation matrices. Andrianomena and Hassan, 2023 analyze latent codes with t-SNE and a linear predictor and assess generated images with power spectra, not descriptive stats or Pearson correlations on latents. Iwasaki et al., 2023 use PCA on latent variables, BIC model comparison, mutual information with physical properties, and SHAP analyses, but no Pearson correlation matrices or basic descriptive stats of latent spaces. Skowronski et al., 2025 presents correlation matrices for R-matrix fit parameters (not latent variables) in a nuclear reaction study. Thus the stated methodology is not supported.
- ✘ **The numerical computation of the latent-space derivatives $\frac{\partial L}{\partial x}$, $\frac{\partial L}{\partial t}$, and $\frac{\partial^2 L}{\partial x^2}$ employs standard finite-difference schemes on uniform grids, specifically second-order central differences for interior points and first-order forward/backward differences at boundaries, consistent with high-order finite-difference and WENO-type methods used in astrophysical and MHD simulations.**
- *Reference(s)*: Caunt and Korpi, 2001, Minoshima et al., 2019, Balsara et al., 2024
- *Justification*: The cited works use high-order finite-difference/WENO schemes, not second-order central with first-order one-sided boundaries. Caunt and Korpi, 2001 employ centered 6th-order finite differences (Appendix A) with a 3rd-order boundary scheme and symmetric/antisymmetric ghosting, not first-order one-sided differences. Minoshima et al., 2019 use a WCNS finite-difference framework with 4th-order central differencing (Eq. 14) and a 3rd-order boundary scheme (Eq. 15), plus high-order WENO interpolation—again not second-order/first-order. Balsara et al., 2024 develop high-order (up to 9th) FD-WENO/AFD-WENO methods; no default second-order central or first-order boundary stencils are advocated. Thus the statement is not supported by these papers.
- ✘ **The construction of the candidate function library for sparse regression is explicitly inspired by the Sparse Identification of Nonlinear Dynamics (SINDy) framework, assembling a matrix Θ that includes constant, linear (in L , $\frac{\partial L}{\partial x}$, $\frac{\partial L}{\partial t}$), and quadratic interaction terms such as $L_j(\frac{\partial L_m}{\partial x})$, which are chosen to mirror PDE structures like those in Burger’s equation.**
- *Reference(s)*: Hatipoğlu, 2022, Hatipoğlu, 2023
- *Justification*: Hatipoğlu, 2022 focuses on PCA/ICA/SSA methods for exoplanet data and does not discuss SINDy, sparse regression, or constructing a SINDy-style candidate function library Θ with constant, linear (in L , V_x , V_{xx}), or quadratic interaction

terms (e.g., $L_j \partial L_m / \partial x$) mirroring PDEs like Burgers'. There is no evidence in the attached works (including Hatipoğlu, 2023) supporting this statement.

- ✘ **Sparse regression is performed using Lasso, minimizing $\|y - \Theta \xi\|_2 + \alpha \|\xi\|_1$ with scores of approximately 0.81–0.93 with 16–24 non-zero terms out of 61 candidates per equation, indicating a balance between accuracy and sparsity consistent with earlier applications of such methods.** with $\alpha = 0.01$, following prior work that uses L_1 -regularized regression to obtain parsimonious models from high-dimensional scientific data, and the resulting latent-space PDEs achieve R^2
- *Reference(s)*: Wohlberg and Wozniak, 2021, Colgan et al., 2022, Wang et al., 2023
- *Justification*: Neither paper discusses Lasso sparse regression for PDE discovery or reports R^2 values and sparsity counts as stated. Wohlberg and Wozniak, 2021 presents convolutional dictionary learning for PSF estimation with an l_1 - l_2 regularizer, not Lasso-based regression or PDE models. Wang et al., 2023 develops a Deep Wiener Deconvolution Network for PSF deconvolution and evaluates with PSNR/SSIM and moment-based metrics, with no L_1 -regularized regression, PDEs, or R^2 results. Therefore, the claim is not supported by the attached papers.

Mathematical consistency audit

This section audits **symbolic/analytic** mathematical consistency (algebra, derivations, dimensional/unit checks, definition consistency).

Maths relevance: substantial

The paper's mathematics is primarily definitional/constructive: it defines a latent embedding $L(x, t) \in \mathbb{R}^{10}$ learned by a PINN, computes finite-difference approximations to $\partial L / \partial x$, $\partial L / \partial t$, and $\partial^2 L / \partial x^2$, analyzes geometry via norms/cosine similarities and an SVD-based tangent-space diagnostic, and proposes a sparse-regression (Lasso) framework to identify latent PDE-like evolution laws $\partial L / \partial t = f(L, V_x, V_{xx})$. Most individual formulas are standard and internally consistent, but the key regression library Θ is described inconsistently between Methods and Results, preventing a fully consistent symbolic audit of the paper's central equation-discovery claims.

Checked items

1. ✓ **Burgers PDE statement** (Eq. (1), Sec. 1, p. 2)

- **Claim:** The governing PDE used for context is $\partial u / \partial t + u \partial u / \partial x - \nu \partial^2 u / \partial x^2 = 0$.
- **Checks:** notation consistency, structural sanity check
- **Verdict:** PASS; confidence: high; impact: moderate
- **Assumptions/inputs:** $u = u(x, t)$ scalar field, ν is viscosity parameter

- **Notes:** Equation is syntactically consistent and matches later references to advection-like $u(\partial u/\partial x)$ and diffusion-like $\partial^2 u/\partial x^2$ operators. Separate terminology issue exists about calling this “2D”.
2. ✓ **Latent space and derivative field definitions** (Sec. 1, p. 2 (definitions of L , V_x , V_t); reiterated Sec. 2.3, p. 3)
- **Claim:** Defines $L(x, t) \in \mathbb{R}^{10}$ and vector fields $V_x = \partial L/\partial x$, $V_t = \partial L/\partial t$, $V_{xx} = \partial^2 L/\partial x^2$.
 - **Checks:** definition consistency, index consistency
 - **Verdict:** PASS; confidence: high; impact: critical
 - **Assumptions/inputs:** L has 10 components L_k , $k = 0, \dots, 9$, x, t are scalar coordinates on a grid
 - **Notes:** Definitions are consistent throughout the Methods/geometry sections. V_{xx} is consistently treated as a componentwise second derivative in x .
3. ✓ **Spatial derivative finite differences (first order)** (Sec. 2.3.1, p. 3)
- **Claim:** Uses central difference (interior) and forward/backward differences (boundaries) to approximate $\partial L_k/\partial x$.
 - **Checks:** algebra, discretization formula sanity check
 - **Verdict:** PASS; confidence: high; impact: moderate
 - **Assumptions/inputs:** Uniform grid with spacing Δx , Indices $i = 0, \dots, N_x - 1$
 - **Notes:** Central difference: $(L_k(x_{i+1}, t_j) - L_k(x_{i-1}, t_j))/(2\Delta x)$ is correct. Boundary one-sided formulas are consistent.
4. ✓ **Temporal derivative finite differences (first order)** (Sec. 2.3.2, pp. 3–4)
- **Claim:** Uses central difference (interior) and forward/backward differences (boundaries) to approximate $\partial L_k/\partial t$.
 - **Checks:** algebra, discretization formula sanity check
 - **Verdict:** PASS; confidence: high; impact: moderate
 - **Assumptions/inputs:** Uniform grid with spacing Δt , Indices $j = 0, \dots, N_t - 1$
 - **Notes:** Formulas match standard centered and one-sided first differences and are internally consistent with indexing.
5. ✓ **Second spatial derivative finite differences** (Sec. 2.3.3, p. 4)
- **Claim:** Approximates $\partial^2 L_k/\partial x^2$ via centered second difference for interior points.
 - **Checks:** algebra, discretization formula sanity check
 - **Verdict:** PASS; confidence: high; impact: moderate

- **Assumptions/inputs:** Uniform grid spacing Δx , Interior indices $1 \leq i \leq N_x - 2$
 - **Notes:** Second derivative formula $(L_{i+1} - 2L_i + L_{i-1})/(\Delta x)^2$ is correct as written.
6. ✓ **Vector magnitude (Euclidean norm) definitions** (Sec. 2.4.1, p. 4)
- **Claim:** Defines $|L|, |V_x|, |V_t|$ as $\sqrt{\sum_{k=0}^9 \text{component}^2}$.
 - **Checks:** definition correctness, index bounds consistency
 - **Verdict:** PASS; confidence: high; impact: minor
 - **Assumptions/inputs:** Latent dimension is 10 with indices 0..9
 - **Notes:** Norm definitions are correct and consistent with 10-dimensional vectors.
7. ✓ **Cosine similarity formula and exclusion rule** (Sec. 2.4.2, p. 4)
- **Claim:** Uses $\cos \theta = (A \cdot B)/(|A||B|)$ and excludes points where a vector magnitude is zero.
 - **Checks:** algebra, well-posedness
 - **Verdict:** PASS; confidence: high; impact: minor
 - **Assumptions/inputs:** Dot product is standard Euclidean, Zero norms cause division by zero
 - **Notes:** Formula is correct; exclusion rule appropriately avoids undefined values.
8. ✓ **Tangent matrix construction for SVD** (Sec. 2.5, p. 4)
- **Claim:** Forms M as a 10×2 matrix with columns V_x and V_t at each (x_i, t_j) , then computes singular values $\sigma_1 \geq \sigma_2 \geq 0$.
 - **Checks:** dimension/shape consistency, linear algebra consistency
 - **Verdict:** PASS; confidence: high; impact: moderate
 - **Assumptions/inputs:** V_x, V_t are 10-vectors at each grid point
 - **Notes:** Shape is consistent: stacking two 10-vectors as columns yields a 10×2 matrix with exactly two singular values.
9. △ **Interpretation of SVD singular values as ellipse axes** (Sec. 2.5, p. 4)
- **Claim:** Claims σ_1, σ_2 represent principal semi-axis magnitudes of the image of the unit circle in (x, t) under the local linear map to latent space.
 - **Checks:** derivation logic, invariance/sensitivity sanity check
 - **Verdict:** UNCERTAIN; confidence: medium; impact: moderate
 - **Assumptions/inputs:** Local linearization $dL \approx V_x dx + V_t dt$, A metric/scale on (dx, dt) is chosen (implicitly Euclidean)

- **Notes:** The statement is conditionally correct if one treats (dx, dt) with a specific Euclidean scaling. Because x and t generally have different units/scales and the paper does not specify nondimensionalization/normalization, σ_2/σ_1 is not invariant to rescaling t and the geometric interpretation can change. Clarification is needed to make this interpretation mathematically well-defined.
10. ✓ **Regression target equation form** (Sec. 2.6 (opening) and Sec. 2.6.3, p. 5)
- **Claim:** For each k , seeks $\partial L_k/\partial t \approx f_k(L, V_x, V_{xx})$, implemented as $V_{t,\text{flat}}[:, k] \approx \Theta \Xi_k$.
 - **Checks:** notation consistency, linear model form
 - **Verdict:** PASS; confidence: high; impact: critical
 - **Assumptions/inputs:** Θ columns are candidate functions evaluated pointwise, Ξ_k is a coefficient vector
 - **Notes:** The linear-in-parameters regression form is consistent with sparse identification frameworks and matches the written $V_{t,\text{flat}} \approx \Theta \Xi_k$ statement.
11. ✓ **Lasso objective function statement** (Sec. 2.6.3, p. 5)
- **Claim:** Minimizes $|y_k - \Theta \Xi_k|_2^2 + \alpha |\Xi_k|_1$.
 - **Checks:** objective correctness, norm notation
 - **Verdict:** PASS; confidence: high; impact: minor
 - **Assumptions/inputs:** $\alpha > 0$ regularization parameter, y_k is the flattened target vector for component k
 - **Notes:** Objective is correctly stated and internally coherent with the preceding regression model.
12. ✘ **Candidate library definition vs reported library in Results** (Sec. 2.6.2, p. 5 (library includes $L_j \partial L_m / \partial x$ for all j, m); Sec. 3.4, p. 9 (library includes $L_j^2, L_j V_{x,j}, V_{x,j}^2$ and totals 61 terms))
- **Claim:** The same Θ is used throughout, with a stated set of candidate terms and a stated total number of candidates.
 - **Checks:** definition consistency, term counting
 - **Verdict:** FAIL; confidence: high; impact: critical
 - **Assumptions/inputs:** Θ used in Sec. 3.4 is the Θ defined in Sec. 2.6.2
 - **Notes:** If Sec. 2.6.2 truly includes all cross terms $L_j(\partial L_m / \partial x)$ for $j, m \in 0, \dots, 9$, that alone contributes 100 quadratic terms, contradicting the later claim of 61 total terms. Conversely, the 61-term count in Sec. 3.4 is consistent with constant + 3×10 linear (L, V_x, V_{xx}) + 3×10 quadratic ($L^2, L \cdot V_x, V_x^2$). The paper must reconcile which library was actually used.
13. △ **Advection/diffusion analogy term notation** (Abstract p. 1; Sec. 3.4, p. 9)

- **Claim:** Identified latent PDEs contain advection-like terms $(L_j V_{x,j})$ and diffusion-like terms $(V_{xx,j})$.
- **Checks:** notation consistency, symbol meaning consistency
- **Verdict:** UNCERTAIN; confidence: medium; impact: minor
- **Assumptions/inputs:** $V_{x,j}$ denotes $\partial L_j / \partial x$ and $V_{xx,j}$ denotes $\partial^2 L_j / \partial x^2$
- **Notes:** The intended componentwise meaning is plausible and consistent with earlier $V_{x,k}$ notation, but the paper alternates between $V_{x,k}$ and $V_{x,j}$ and uses comma subscripts without an explicit convention. Clarifying that $V_{x,j} := \partial L_j / \partial x$ would remove ambiguity.

14. ✓ **Array reshaping and dimensional consistency for regression** (Sec. 2.6.1, p. 5)

- **Claim:** Reshapes $(N_x, N_t, 10)$ arrays into $(N_x \cdot N_t, 10)$ flattened arrays for regression.
- **Checks:** dimension/shape consistency
- **Verdict:** PASS; confidence: high; impact: minor
- **Assumptions/inputs:** $N_x = N_t = 100$, Flattening preserves correspondence between features and targets at each grid point
- **Notes:** Shape transformation is consistent with standard supervised regression preparation; symbolic consistency is fine.

Limitations

- Only one numbered equation (Eq. (1)) is explicitly provided; the discovered latent PDE system is not written out fully, so its algebraic term-by-term correctness cannot be audited.
- Figures are referenced for empirical distributions, but the audit does not evaluate numeric values; it only checks that the associated formulas are well-defined.
- The paper does not provide an explicit column-by-column definition of Θ or its exact ordering; combined with the Methods/Results inconsistency, verification of the sparse-identification pipeline is blocked at the symbolic specification level.

Numerical results audit

This section audits **numerical/empirical** consistency: reported metrics, experimental design, baseline comparisons, statistical evidence, leakage risks, and reproducibility.

Executed 11 numeric checks: 10 PASS and 1 UNCERTAIN due to an execution error. Passed checks support consistency of channel counts, grid-size products, library term-count recomputation to 61, sparsity percentage calculations (16/61 and 24/61), multiple inequality/range constraints, and threshold claims for example correlations. One reshape element-count check for flattening arrays to $(N_x \times N_t, 10)$ could not be verified by the automated checker.

Checked items

1. ✓ **C1** (Section 2.1 (page 3): “dataset ... dimensions (100,100,12) ... first two channels ... remaining 10 channels”)
 - **Claim:** The dataset has shape (100,100,12) where 2 channels are (x, t) and the remaining 10 channels are latent dimensions.
 - **Checks:** dimension_consistency (channels sum)
 - **Verdict:** PASS
 - **Notes:** Checked coord_channels + latent_channels against total_channels.
2. ✓ **C2** (Section 2.1 (page 3): “grid of 100×100 points ... $N_x = 100$... $N_t = 100$... reshape ... (10000,10)”)
 - **Claim:** $N_x \times N_t$ equals 10000 (used for reshaping latent space to (10000,10)).
 - **Checks:** product_consistency
 - **Verdict:** PASS
 - **Notes:** Checked product $N_x \times N_t$ against flattened_rows.
3. ⚠ **C3** (Section 2.6.1 (page 5): “flattening ... arrays of shape $(N_x \times N_t, 10)$ ”)
 - **Claim:** Flattened arrays from (100,100,10) should become (10000,10).
 - **Checks:** reshape_consistency
 - **Verdict:** UNCERTAIN
 - **Notes:** Exception during check: TypeError: type str doesn't define **round** method
4. ✓ **C4** (Section 2.6.2 (page 5) vs Section 3.4 (page 9): library term count “out of 61 total candidate terms”)
 - **Claim:** Total number of candidate terms in Θ should equal 61 given the stated included term types.
 - **Checks:** term_count_recomputation
 - **Verdict:** PASS
 - **Notes:** Recomputed total terms using stated breakdown $1 + 6 \times \text{latent_dims_j}$.
5. ✓ **C5** (Section 3.4 (page 9): “non-zero terms ... 16 to 24 out of 61 ... sparsity levels between 26% and 39%”)
 - **Claim:** Sparsity percentages should match counts/total: $16/61 \approx 26\%$ and $24/61 \approx 39\%$.
 - **Checks:** percentage_from_counts
 - **Verdict:** PASS
 - **Notes:** Compared computed fractions min_nonzero/total and max_nonzero/total to reported percentages.

6. ✓ **C6** (Section 3.3.3 (page 8): “mean σ_1 is 3.71 ... mean σ_2 is 0.45 ... ratio σ_2/σ_1 has a mean of 0.14”)
- **Claim:** Check whether mean ratio (0.14) is consistent with the ratio of means $0.45/3.71 \approx 0.12$ (not necessarily equal, but should be plausibly close).
 - **Checks:** derived_ratio_sanity
 - **Verdict:** PASS
 - **Notes:** Sanity check: compared reported $\text{mean}(\sigma_2/\sigma_1)$ to $\sigma_2\text{mean}/\sigma_1\text{mean}$ (not expected to match exactly).
7. ✓ **C7** (Section 3.1 (page 6): “variance ... minimum ≈ 0.48 for L_2 to maximum ≈ 1.95 for L_3 ”)
- **Claim:** Variance range endpoints are ordered correctly and can be checked for basic consistency (min < max).
 - **Checks:** inequality_sanity
 - **Verdict:** PASS
 - **Notes:** Checked strict ordering min < max.
8. ✓ **C8** (Section 3.1 (page 6): correlations examples “ L_0 with L_3 (-0.96)... L_2 with L_4 (0.89) and L_8 (0.88)... abs values > 0.8”)
- **Claim:** Example correlations claimed to have $|r| > 0.8$ should satisfy that threshold numerically.
 - **Checks:** threshold_check
 - **Verdict:** PASS
 - **Notes:** Checked each example correlation exceeds the stated threshold in absolute value.
9. ✓ **C9** (Figure 3 caption / Section 3.1 (page 7): “Pearson correlation coefficient ≈ -0.45 between L_0 and L_4 ”)
- **Claim:** The correlation value -0.45 should be within $[-1, 1]$ and sign matches 'moderate negative'.
 - **Checks:** range_check
 - **Verdict:** PASS
 - **Notes:** Checked all provided values lie within $[-1, 1]$.
10. ✓ **C10** (Section 3.3.1 (page 7): “mean magnitudes ... 2.82 for $|L|$, 3.52 for $|V_x|$, 0.93 for $|V_t|$ ”)
- **Claim:** Claimed ordering: $\text{mean}(|V_x|) > \text{mean}(|L|) > \text{mean}(|V_t|)$.
 - **Checks:** inequality_sanity
 - **Verdict:** PASS
 - **Notes:** Checked strict inequality chain.

11. ✓ **C11** (Section 3.3.2 (page 7): cosine similarity stats “mean near zero (0.125)” and “mean (0.237)” and “mean -0.193 median -0.496 ”)
- **Claim:** Cosine similarity means/medians must lie in $[-1, 1]$.
 - **Checks:** `range_check`
 - **Verdict:** PASS
 - **Notes:** Checked all provided values lie within $[-1, 1]$.

Limitations

- Only parsed text from the provided PDF was used; no external data, code repositories, or internet sources were consulted.
- Checks that require access to the underlying NumPy arrays (`Lspace`, `Vx`, `Vt`, `Vxx`) or regression outputs cannot be executed from the PDF alone and are listed as unverified.
- Numerical values embedded only in figures/histograms are not extracted because plot/pixel-based value extraction is out of scope.
- Execution error encountered during automated checking: C3: TypeError: type str doesn't define **round** method.