

Skeptical review: Analyzing the Local Intrinsic Dimension of Physics-Informed Neural Network Latent Spaces for Burger’s Equation

Summary

The paper proposes using Local Intrinsic Dimension (LID) to probe how a Physics-Informed Neural Network (PINN) encodes solutions to Burgers’ equation in a 10-dimensional hidden/latent layer. From a pre-trained PINN evaluated on a 100×100 (x, t) grid, the authors extract 10,000 latent vectors $z(x, t)$, estimate pointwise LID via a k NN distance-scaling log-log regression for $k \in [5, 20]$, and reshape the results into a spatio-temporal field $D(x, t)$. The empirical distribution is strongly low-dimensional on average (reported mean ≈ 1.88) but heterogeneous, with coherent bands in $D(x, t)$ that are interpreted as aligning with different physical regimes (e.g., shocks vs. smooth regions). The methodology is promising as an interpretability diagnostic, but the current manuscript remains largely exploratory: the Burgers/PINN setup and solution quality are under-specified, the central physical interpretation is not quantitatively validated against $u(x, t)$ or its derivatives, and robustness/reproducibility of the LID estimates (including extreme values exceeding the 10D embedding) is not sufficiently assessed.

Strengths

- Clear high-level pipeline from latent extraction to k NN-based LID estimation and reconstruction of a spatio-temporal map $D(x, t)$ (Sec. 2.1–2.3).
- Novel framing of LID as a physics-indexed descriptor field ($D(x, t)$) for analyzing internal representations of PINNs, which could be broadly useful for interpretability (Sec. 1, Sec. 2.3.3, Sec. 3.3–3.4).
- Provides useful descriptive statistics of the LID distribution (mean/median/std/range) showing strong apparent dimensionality reduction relative to the 10D embedding (Sec. 3.2–3.3).
- The transformation from the power-law scaling $r_k(z_p) \propto k^{1/D_p}$ to a log-linear regression and estimation of D_p as $1/m_p$ is algebraically consistent (Sec. 2.3.2).
- Visualizations (histogram and heatmap) effectively convey that LID varies strongly across the (x, t) domain and forms coherent structures rather than noise (Fig. 1–2, Sec. 3.3).
- Indexing/reshaping between the flattened point index p and the 100×100 grid appears internally consistent, making the construction of $D(x, t)$ actionable to reproduce (Sec. 2.1, Sec. 2.3.3).

Major issues

1. **The PDE problem definition, the PINN architecture/training, and the quality/accuracy of the learned solution are insufficiently specified, limiting interpretability and reproducibility (Sec. 1, Sec. 2.1, Sec. 3).** The manuscript refers to “2D Burgers” while using a 2D (x, t) grid, but does not explicitly state the governing equation (viscous vs. inviscid; ν value; any forcing), domain, and initial/boundary conditions. Likewise, key PINN details are missing (layer sizes/activations; where the 10D “latent layer” is taken—pre/post nonlinearity; loss terms and weights; collocation/boundary sampling; optimizer/schedule/epochs; checkpoint used for latent extraction). Without validation (e.g., error vs. reference solver; PDE residual maps), it is unclear whether observed LID patterns reflect physics, architecture choices, or training artifacts.

Recommendation: Add a dedicated setup section (expand Sec. 2.1 or add a new Sec. 2.0) that (i) writes the exact Burgers’ equation solved, including ν (and forcing if any), domain, and initial/boundary conditions; (ii) clarifies that this is 1D-in-space Burgers’ with time on a 2D spatio-temporal grid (or explicitly states if truly 2D in space); (iii) fully specifies the PINN (layers/widths/activations; location/definition of the 10D latent vector; loss components and weights; training data/collocation strategy; optimizer and stopping criteria; random seed). In Sec. 3, include basic solution-quality evidence: plots of $u(x, t)$ and PDE residual, plus L^2 /relative error against a standard numerical reference (or at least residual/BC-IC satisfaction metrics). State precisely which checkpoint produced the analyzed latent vectors.

2. **The central physical interpretation (“low LID bands correspond to shocks/high complexity; higher LID corresponds to smooth regions”) is currently qualitative and not directly validated against physical fields (Sec. 3.3–3.4, Sec. 4).** The paper does not present $u(x, t)$ itself, nor gradient/curvature/shock indicators (e.g., $|\partial u/\partial x|$, $|\partial^2 u/\partial x^2|$, total variation, entropy production), nor quantitative association tests. This leaves the main claim speculative.

Recommendation: Augment Sec. 3.3–3.4 with quantitative comparisons between $D(x, t)$ and physics-derived fields: (1) plot $u(x, t)$ and derived measures such as $|\partial u/\partial x|$ (and optionally $|\partial^2 u/\partial x^2|$, $|\partial u/\partial t|$, PDE residual magnitude) alongside $D(x, t)$ at representative times; (2) define a “shock/steep-layer region” via a standard threshold on $|\partial u/\partial x|$ or total variation and compare the LID distributions inside vs. outside; (3) report correlation/MI/scatter analyses between $D(x, t)$ and these complexity metrics over all grid points. If the solution is viscous (finite ν), rephrase ‘shock’ as ‘steep gradient layer’ and interpret accordingly. Update Sec. 4 conclusions to separate validated findings from hypotheses.

3. **Robustness and reliability of the LID estimates are not established, despite extreme values (LID < 1 and LID > 10, with maxima ≈ 14.4 exceeding the 10D embedding) and potential sensitivity to k -range, metric, anisotropy, and regression fit quality (Sec. 2.3.1–2.3.2, Sec. 3.2–3.3).** Using Euclidean distance on raw latent coordinates together with a narrow k range (5–20) can yield unstable estimates, particularly if latent dimensions have very different scales/heavy tails (as suggested by the EDA). The manuscript also does not report regression goodness-of-fit (e.g., R^2) or uncertainty for D_p .

Recommendation: Extend Sec. 2.3 and Sec. 3.2–3.3 with robustness/diagnostics: (1) sensitivity to k choices (e.g., [3, 10], [5, 15], [10, 30]) and report how global stats and spatial patterns of $D(\mathbf{x}, t)$ change; (2) compare distances computed on raw \mathbf{z} vs. standardized/whitened \mathbf{z} (and state the chosen preprocessing); (3) report regression diagnostics per point (R^2 distribution, slope m_p distribution) and flag or mask unreliable fits (e.g., low R^2); (4) consider a second estimator on a subset (e.g., Levina–Bickel MLE LID, TWO-NN) or bootstrap/subsampling to quantify variance. In Sec. 3.3–3.4, explicitly interpret extreme values as potential estimator/finite-sample artifacts unless shown robust, and report the fraction of points with LID > 10 or LID < 1.

4. **Methodological/implementation details are not sufficient for faithful reproduction of the LID pipeline and latent extraction (Sec. 2.2–2.3).** Critical ambiguities include: kNN implementation and parameters; whether the query point is included as its own nearest neighbor; how ties/duplicate distances and near-zero distances are handled before taking logs; the log base; the exact regression routine; and whether any filtering (e.g., $m_p \leq 0$, NaNs) was necessary. The text also mentions possible negative slopes ($m_p < 0$), which should not occur if r_k is defined as the k -th nearest-neighbor distance with k increasing and distances sorted, suggesting either a definitional mismatch or an implementation detail that must be clarified (Sec. 2.3.2).

Recommendation: In Sec. 2.3.1–2.3.2, add implementation-level specifics (and optionally short pseudocode): name the library/function used for kNN (e.g., sklearn NearestNeighbors), metric, whether self-neighbors are excluded, and how r_k and k are indexed; specify log base; define handling of $r_k = 0$ or ties (epsilon, jitter, or skipping); specify the regression implementation (e.g., numpy.polyfit/OLS) and whether any robust fitting is used. Correct the $m_p < 0$ discussion: under the stated definition it should be $m_p \geq 0$; if negatives occurred, explain what differs (e.g., unsorted distances, including self, numerical issues). Report NaN counts and any validity filtering explicitly in Sec. 3.2.

5. **The manuscript occasionally overreaches from a geometric descriptor (LID of a latent point cloud) to causal claims about “adaptive compression strategies” and broader generality, while evidence comes from a single model/run and a single PDE setting (Sec. 3.5, Sec. 4).** Low LID does not uniquely imply compression in an information-theoretic or capacity-allocation sense; it can also arise from activation saturation, symmetries/degeneracies, sampling density issues, or layer choice.

Recommendation: Reframe Sec. 3.5 and Sec. 4 to clearly distinguish observation from interpretation (e.g., ‘consistent with’ rather than ‘demonstrates’). Add at least one minimal generality check where feasible: compute $D(\mathbf{x}, t)$ for (i) a different hidden layer (layer-wise LID), and/or (ii) another seed or slight architecture/latent-dimension variant. If possible, complement LID with a Jacobian-based local rank/effective-rank measure (e.g., rank of $\partial \mathbf{z} / \partial \mathbf{x}(\mathbf{x}, t)$) to strengthen the ‘compression/representation’ narrative. Expand related work in Sec. 1 to better situate the contribution within PINN interpretability and intrinsic-dimension analysis in deep networks.

Minor issues

1. Ambiguity in terminology: “2D Burgers equation” is likely meant as a 2D spatio-temporal (\mathbf{x}, t) grid for a 1D spatial PDE, but it can be read as two spatial dimensions (Sec. 1–2, Sec. 4).

Recommendation: Clarify early (Sec. 1 and the start of Sec. 2.1) whether Burgers’ is 1D-in-space with time on a 2D grid, or truly 2D in space. Use consistent phrasing throughout (e.g., “1D Burgers’ equation on an (\mathbf{x}, t) grid”).

2. The mapping between stored data and physical/latent quantities is unclear: the array is described as (100, 100, 12) with 10 latent slices, but it is not explicit where $\mathbf{u}(\mathbf{x}, t)$, \mathbf{x} , and t are stored and how they align with latent vectors (Sec. 2.1). This also blocks direct $D(\mathbf{x}, t)$ vs $\mathbf{u}(\mathbf{x}, t)$ analyses.

Recommendation: In Sec. 2.1, document the exact array semantics (which channel indices correspond to \mathbf{x} , t , \mathbf{u} , and the 10 latent components) or explicitly state that $\mathbf{u}(\mathbf{x}, t)$ is sourced elsewhere. Add a one-line definition linking D_p to the grid, e.g., $D(\mathbf{x}_i, t_j) = D_{p(i,j)}$ with $p = i \times 100 + j$.

3. Exploratory data analysis (EDA) on latent coordinates is mentioned but underreported, and some interpretive statements about individual latent dimensions are not well supported (Sec. 2.2, Sec. 3.1). Table 1 is referenced but appears missing.

Recommendation: Ensure Table 1 is included (or inline the key numbers). Specify exactly how skewness/kurtosis are computed (library/function and bias correction). If making claims about distinct roles of latent dimensions, add minimal supporting analyses (pairwise correlations, PCA explained variance, or simple linear probes) or tone down to hypotheses.

4. Figure 1 and Figure 2 would be more actionable with additional annotations and clearer communication of estimator settings/uncertainty (Fig. 1–2; Sec. 3.2–3.3). As-is, key statistics (mean/median, embedding dimension = 10, tail fractions) and estimator parameters (k range, metric, preprocessing) are not visually prominent.

Recommendation: Fig. 1: add vertical lines for mean/median and a marker at 10; include $n = 10,000$; state binning in the caption; consider a linear-scale inset for $LID \in [0, 5]$ and annotate tail proportions (e.g., % with $LID > 10$). Fig. 2: label axes with units/ranges and include k -range/metric/preprocessing in the caption; consider rescaling/clipping or adding an inset to highlight the 0–4 range; if available, add an uncertainty visualization (e.g., bootstrap SE map) and/or overlay contours of $|\partial u / \partial x|$ once computed.

5. Related work is relatively narrow given the paper’s intended contribution to PINN interpretability and representation analysis (Sec. 1).

Recommendation: Expand Sec. 1 to cite broader PINN literature and prior work on intrinsic dimension/representation geometry in deep networks; more explicitly state what is new here (constructing a physics-indexed LID field from a PINN latent layer) relative to existing intrinsic-dimension analyses.

6. The conclusion could better balance contributions with limitations and concrete next steps (Sec. 4).

Recommendation: Add a short limitations paragraph (single-PDE/single-model; estimator sensitivity; lack of uncertainty; currently qualitative physics linkage) and then list 2–4 specific next experiments (layer-wise LID; multi-seed/architecture; alternative LID estimators; quantitative correlation with gradients/residual).

7. Non-technical presentation mismatches: keywords/contextual phrasing emphasize astronomy terms that do not appear central to the Burgers/PINN LID study (Abstract, Sec. 1).

Recommendation: Align keywords with the actual content (PINNs, Burgers’ equation, intrinsic dimension, latent space). If the venue requires astronomy framing, motivate it explicitly (e.g., intended applications to astrophysical fluid dynamics).

Very minor issues

1. Inconsistent naming/typography and formatting reduce polish (Abstract, Sec. 1–4): “Burger’s” vs. standard “Burgers’/Burgers”; stray backticks/quotes around array slices; occasional markdown-style headings (e.g., leading ‘#’); awkward line breaks splitting words.

Recommendation: Proofread and standardize terminology (“Burgers’ equation”), heading styles, and code/array formatting (LaTeX verbatim or consistent monospace). Fix line-break hyphenation artifacts.

2. Minor notation inconsistency: $D(x, t)$, D_{map} , and D_p are used without a single explicit defining equation tying them together (Sec. 2.3.3).

Recommendation: Add one explicit definition in Sec. 2.3.3 (e.g., $D(x_i, t_j) := D_{p(i,j)}$ with $p = i \times 100 + j$) and then use one preferred symbol consistently.

3. The manuscript notes that positive slopes occur for all points but does not explicitly report counts of NaNs/invalid fits (Sec. 3.2).

Recommendation: Add a one-line audit in Sec. 3.2: number of points, number excluded (if any), NaN count, and summary of slope positivity after filtering (if filtering is applied).

Key statements and references

- ✓ **The concept of Intrinsic Dimension (ID) provides a measure of the true dimensionality of the manifold on which data points lie, which can be significantly lower than the embedding dimension, and the Local Intrinsic Dimension (LID) extends this concept to estimate the effective dimensionality in the neighborhood of individual data points, enabling characterization of how manifold complexity varies across space.**
- *Reference(s):* Candelori et al., 2024, Cadiou et al., 2025, Erba et al., 2020
- *Justification:* ID is defined as the manifold’s true dimensionality d (often $d \ll D$) and the minimal number of parameters to characterize the data (Candelori et al., 2024; Erba et al., 2020). Both Candelori et al. and Erba et al. provide neighborhood-based/local estimates: Candelori et al. return a list of local intrinsic dimension estimates $d_x \approx \{\text{rm dim}\}x$ *Mforeachpointandshowspatialvariability(e.g., MNIST digit‘1’)*, while Erba et al. introduce a multiscale method that computes local ID $d(r_c)$ from a point’s neighbors, revealing varying complexity across regions (e.g., Swiss roll, intersecting manifolds). Cadiou et al., 2025 also note that many intrinsic-dimension methods are local. This directly supports the statement about ID and its local (LID) extension.
- ✘ **To construct the $D(x, t)$ map, the study employs a k -nearest neighbor based regression method for LID estimation in which, for each point z_p , Euclidean distances to its k -th nearest neighbors are computed and the LID D_p is obtained as the reciprocal of the slope from an ordinary least squares linear regression of $\log(r_k(z_p))$ against $\log(k)$ over a chosen k -range, following standard LID estimation practice.**
- *Reference(s):* Luken et al., 2018, Han et al., 2020, Luken et al., 2018

- *Justification*: None of the attached papers construct a $D(\mathbf{x}, t)$ map or perform local intrinsic dimensionality (LID) estimation via OLS of log neighbor distances. Both Luken et al., 2018 and Han et al., 2020 use k NN for photometric redshift prediction (with Euclidean/Manhattan distances and mean or weighted mean/median of neighbor redshifts), not LID estimation. Therefore the stated method is not supported by these papers.
- **✘ In estimating LID, it is theoretically expected that the regression slope m_p relating $\log(r_k(z_p))$ to $\log(k)$ is positive for data residing on a manifold; however, prior work has shown that in finite datasets or regions with complex structure the regression can yield non-positive slopes ($m_p \leq 0$), in which case the local structure does not conform to the expected scaling behavior and the corresponding LID estimates should be treated as unreliable (e.g., set to NaN).**
- *Reference(s)*: Erba et al., 2020, Özçoban et al., 2025, Erba et al., 2020
- *Justification*: None of the attached papers discuss the LID regression slope m_p from $\log r_k(z_p)$ vs $\log k$ or the possibility of non-positive slopes. Erba et al., 2020 focus on correlation integral/FCI estimators (global and multiscale) and note undersampling-induced bias but do not mention $m_p \leq 0$ or setting estimates to NaN. Özçoban et al., 2025 present a projective, eigenvalue-based estimator, not LID via neighbor-scaling. Therefore the specific claim about non-positive local regression slopes and treating such LID estimates as unreliable is not supported by these papers.

Mathematical consistency audit

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

Maths relevance: light

The paper’s mathematics primarily defines a local intrinsic dimension (LID) estimator using k -nearest neighbor distances in a 10D latent space, derives a log-linear regression form from an assumed power-law scaling, and defines reshaping/indexing operations to map pointwise LID estimates back onto a 100×100 (\mathbf{x}, t) grid. There are no extended derivations beyond the log transformation and slope-to-dimension inversion; the main audit points are algebraic correctness of these transformations and consistency/clarity of definitions (especially the definition of r_k and slope sign).

Checked items

- ✓ **Latent tensor slicing and reshape to Z_{flat}** (Sec. 2.1, p.2-3)
 - **Claim:** Raw data has shape $(100, 100, 12)$, where first two channels are \mathbf{x} and t meshes and remaining 10 channels form $z(\mathbf{x}, t) \in \mathbb{R}^{10}$; reshaping yields Z_{flat} of shape $(10000, 10)$.
 - **Checks:** definition consistency, dimensionality/shape consistency
 - **Verdict:** PASS; confidence: high; impact: minor
 - **Assumptions/inputs:** The last axis ordering is exactly $[\mathbf{x}, t, z_1, \dots, z_{10}]$. Flattening is done over the first two axes ($100 \times 100 = 10000$ points).
 - **Notes:** Shapes are internally consistent: $(100, 100, 10)$ reshapes naturally to $(10000, 10)$.
- ✓ **Flattened index mapping $p = i \times 100 + j$** (Sec. 2.1, p.3)
 - **Claim:** The mapping from grid indices (i, j) to flattened index p is $p = i \times 100 + j$, with inverse $i = p // 100$ and $j = p \% 100$.
 - **Checks:** algebra, definition consistency
 - **Verdict:** PASS; confidence: high; impact: minor
 - **Assumptions/inputs:** Row-major flattening with the second index varying fastest., i and j each range over 0..99.
 - **Notes:** The inverse operations correctly recover i and j under the stated mapping.
- △ **Definition of $r_k(z_p)$ as k -th nearest neighbor distance** (Sec. 2.3 and 2.3.1, p.3)
 - **Claim:** $r_k(z_p)$ denotes the Euclidean distance from z_p to its k -th nearest neighbor within Z_{flat} .
 - **Checks:** definition consistency, well-posedness
 - **Verdict:** UNCERTAIN; confidence: medium; impact: moderate
 - **Assumptions/inputs:** The neighbor distances are ordered so that $r_1 \leq r_2 \leq \dots \leq r_{K_{\text{max}}}$. The query point is excluded from its own neighbor set (not stated).
 - **Notes:** The paper does not specify whether the query point itself is included among “nearest neighbors.” Inclusion changes r_k indexing (and can create $r_1 = 0$). This does not break the algebra shown, but it changes the exact meaning of $r_k(z_p)$ used in the regression.
- ✓ **Power-law scaling to log-linear form** (Sec. 2.3, p.3)
 - **Claim:** From $r_k(z_p) \propto k^{1/D_p}$, it follows that $\log(r_k(z_p)) \approx (1/D_p) \log(k) + C$.
 - **Checks:** algebra, log transform consistency
 - **Verdict:** PASS; confidence: high; impact: critical

- **Assumptions/inputs:** $r_k(z_p)$ is positive for k in the regression range., The proportionality constant does not depend on k over the fitted range.
 - **Notes:** Log transformation of a power law is correct: $\log r = (1/D) \log k + \log \text{const.}$
5. ✓ **Regression model specification** (Sec. 2.3.2, p.3)
- **Claim:** An OLS regression is fit: $\log(r_k(z_p)) = m_p \cdot \log(k) + c_p$ for $k = 5..20$.
 - **Checks:** notation consistency, model form consistency
 - **Verdict:** PASS; confidence: high; impact: moderate
 - **Assumptions/inputs:** The regression is performed pointwise in z_p using the 16 (k, r_k) pairs., $r_k(z_p) > 0$ for all k used.
 - **Notes:** The regression form matches the prior log-linear relationship (with m_p corresponding to $1/D_p$ under the stated scaling).
6. ✓ **Dimension estimate $D_p = 1/m_p$** (Sec. 2.3 and 2.3.2, p.3)
- **Claim:** The LID estimate is computed as $D_p = 1/m_p$.
 - **Checks:** algebra, definition consistency
 - **Verdict:** PASS; confidence: high; impact: critical
 - **Assumptions/inputs:** m_p approximates $1/D_p$ from the scaling law., $m_p > 0$ for accepted estimates.
 - **Notes:** This is the correct inversion given $\log(r) \approx (1/D) \log(k) + C$.
7. ✗ **Claim that m_p may be negative under stated definitions** (Sec. 2.3.2, p.3)
- **Claim:** The regression might yield a non-positive slope ($m_p \leq 0$), including potentially negative values, in practice.
 - **Checks:** analytic sanity check, monotonicity implication
 - **Verdict:** FAIL; confidence: high; impact: minor
 - **Assumptions/inputs:** $r_k(z_p)$ is the ordered k -th nearest-neighbor distance as defined., k is used in increasing order.
 - **Notes:** With $r_k(z_p)$ nondecreasing in k by definition, $y = \log(r_k)$ is nondecreasing with $x = \log(k)$ increasing, which implies an OLS slope cannot be negative. Only $m_p = 0$ is plausible in degenerate/tied-distance cases. Negative m_p would indicate a deviation from the stated r_k definition or data ordering.
8. ✓ **Construction of $D(x, t)$ map from D_p** (Sec. 2.3.3, p.4)
- **Claim:** D_p values are placed into a 100×100 array via $i = p//100, j = p\%100$ so that $\text{Dmap}[i, j] = D_p$ corresponds to $D(x, t)$ on the original grid.
 - **Checks:** indexing consistency, definition consistency
 - **Verdict:** PASS; confidence: high; impact: moderate
 - **Assumptions/inputs:** Same flattening convention used in both reshape and reconstruction.
 - **Notes:** Given the earlier mapping, the reconstruction is consistent and yields an aligned $D(x, t)$ field.
9. △ **Use of logarithms on distances** (Sec. 2.3 and 2.3.2, p.3)
- **Claim:** Compute $\log(r_k(z_p))$ and regress against $\log(k)$.
 - **Checks:** domain of functions, dimensional/unit consistency
 - **Verdict:** UNCERTAIN; confidence: medium; impact: minor
 - **Assumptions/inputs:** $r_k(z_p) > 0$ for $k = 5..20$., Distances in latent space are dimensionless or normalized.
 - **Notes:** Positivity is likely if self is excluded and $k \geq 1$, but the self-inclusion convention is not stated. Also, the paper does not state whether latent-space distances are dimensionless; strictly, logs require dimensionless arguments.

Limitations

- The document provides no explicit equation for Burger's/Burgers' PDE, so no consistency check of PDE formulation, boundary/initial conditions, or nondimensionalization is possible from the provided text.
- The key scaling assumption $r_k(z_p) \propto k^{1/D_p}$ is asserted without a derivation or explicit conditions of validity; this audit only checked the algebraic manipulation that follows from the assumption.
- Because the audit is symbolic/analytic only, it does not evaluate estimator bias/variance, finite-sample effects, or whether the chosen k -range is theoretically appropriate.

Numerical results audit

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

Numeric self-consistency checks for grid sizing/reshaping, raw tensor slice accounting, flattening index bounds, KNN k -range sizing/feasibility, and LID summary rounding/ordering all pass (C1–C9). One claim about absence of NaNs based on positive regression slopes cannot be recomputed without underlying arrays (C10 UNCERTAIN).

Checked items

1. ✓ **C1** (p.1 Abstract; p.2 §1; p.2 §2.1; p.3 §2.1)
 - **Claim:** Full set of 10,000 latent vectors sampled on a 100×100 grid; data reshaped to (10000,10).
 - **Checks:** parts_vs_total / shape-consistency
 - **Verdict:** PASS
 - **Notes:** Verified $100 \times 100 = 10,000$ and element counts match for reshape $(100, 100, 10) \rightarrow (10,000, 10)$; parsed shape equals (10000,10).
2. ✓ **C2** (p.2 §2.1 Data Acquisition and Preparation)
 - **Claim:** Raw NumPy data has shape (100,100,12): first two slices are x and t ; subsequent 10 slices are latent vector ($2 + 10 = 12$).
 - **Checks:** dimension-sum consistency
 - **Verdict:** PASS
 - **Notes:** Confirmed $\text{coord_slices} + \text{latent_slices} = 2 + 10 = 12$.
3. ✓ **C3** (p.3 §2.1)
 - **Claim:** Flattened index mapping $p = i \times 100 + j$, with i, j in $[0, 99]$, yields p from 0 to 9999.
 - **Checks:** index-range consistency
 - **Verdict:** PASS
 - **Notes:** Computed $p_{\min} = 0 \times 100 + 0 = 0$ and $p_{\max} = 99 \times 100 + 99 = 9999$; matches claim.
4. ✓ **C4** (p.3 §2.3.1; p.3 §2.3.2; p.4 §3.2)
 - **Claim:** KNN regression uses k from $k_{\min} = 5$ to $K_{\max} = 20$ (inclusive), implying 16 k -values per regression.
 - **Checks:** range-size consistency
 - **Verdict:** PASS
 - **Notes:** Inclusive integer count $(20 - 5 + 1 = 16)$.
5. ✓ **C5** (p.3 §2.3.1; p.3 §2.3.2)
 - **Claim:** Distances computed to first K_{\max} nearest neighbors with $K_{\max} = 20$; regression uses $k = 5..20$ (requires at least 20 neighbors).
 - **Checks:** parameter-feasibility consistency
 - **Verdict:** PASS
 - **Notes:** Verified $k_{\text{used, max}} \leq K_{\max}$ ($20 \leq 20$).
6. ✓ **C6** (p.4 §3.3)
 - **Claim:** Mean LID reported as 1.8834 and described as approximately 1.88.
 - **Checks:** rounding-consistency
 - **Verdict:** PASS
 - **Notes:** Standard rounding of 1.8834 to 2 decimals gives 1.88.
7. ✓ **C7** (p.4 §3.3; p.1 Abstract; p.6 §3.5; p.6 Conclusions)
 - **Claim:** Mean LID (~ 1.88) is far below embedding dimension 10 (significant dimensionality reduction).
 - **Checks:** unit-consistent comparison
 - **Verdict:** PASS
 - **Notes:** Confirmed $\text{mean_LID} < \text{embedding_dim}$; computed ratio $\text{mean_LID}/\text{embedding_dim} = 0.18834$ (no target asserted beyond inequality).
8. ✓ **C8** (p.4 §3.3; p.6 §3.5; p.6 Conclusions)
 - **Claim:** LID summary stats: mean 1.8834, median 1.6858, std 0.9156, min 0.4723, max 14.4030; textual rounded variants: std ~ 0.92 , min ~ 0.47 , max ~ 14.4 .
 - **Checks:** rounding-consistency + ordering-consistency
 - **Verdict:** PASS
 - **Notes:** Verified $\text{min} \leq \text{median} \leq \text{max}$ and that $\text{round}(\text{std}, 2) = 0.92$, $\text{round}(\text{min}, 2) = 0.47$, $\text{round}(\text{max}, 1) = 14.4$.
9. ✓ **C9** (p.4 §3.3; p.6 §3.5; p.6 Conclusions)
 - **Claim:** Some LID estimates exceed embedding dimension: max LID 14.4030 > 10 .
 - **Checks:** threshold comparison
 - **Verdict:** PASS
 - **Notes:** Confirmed $\text{max_LID} > \text{embedding_dim}$ ($14.403 > 10$).

10. Δ C10 (p.4 §3.2)

- **Claim:** Regression produced positive slopes ($m_p > 0$) for all points; thus no NaN LID values among 10,000 vectors.
- **Checks:** logical implication check (count consistency)
- **Verdict:** UNCERTAIN
- **Notes:** Underlying slope/LID arrays are not available here, so NaN_count== 0 and $m_p > 0$ -for-all cannot be recomputed from provided numerics alone.

Limitations

- Only parsed text was available; Table 1 is missing and figures cannot be quantitatively audited without underlying data.
- No raw arrays (Z_{flat} , distances r_k , slopes m_p , D_p/D_{map}) are provided in the PDF text, so claims about counts of NaNs/positivity of slopes and distributional properties cannot be directly recomputed here.
- Checks are limited to arithmetic/rounding/parameter-range/logical-consistency validations using explicit numerals present in the document.
- One check (C10) is uncertain because the logical implication about NaN counts requires access to computed slope/LID arrays.