EuroVis Workshop on Visual Analytics (2026)
D. Archambault and A. Vilanova (Editors)

Local Neighborhood Instability in Parametric Projections: Quantitative and Visual Analysis

Frederik L. Dennig[1] and Daniel A. Keim[1]

[1]University of Konstanz, Germany

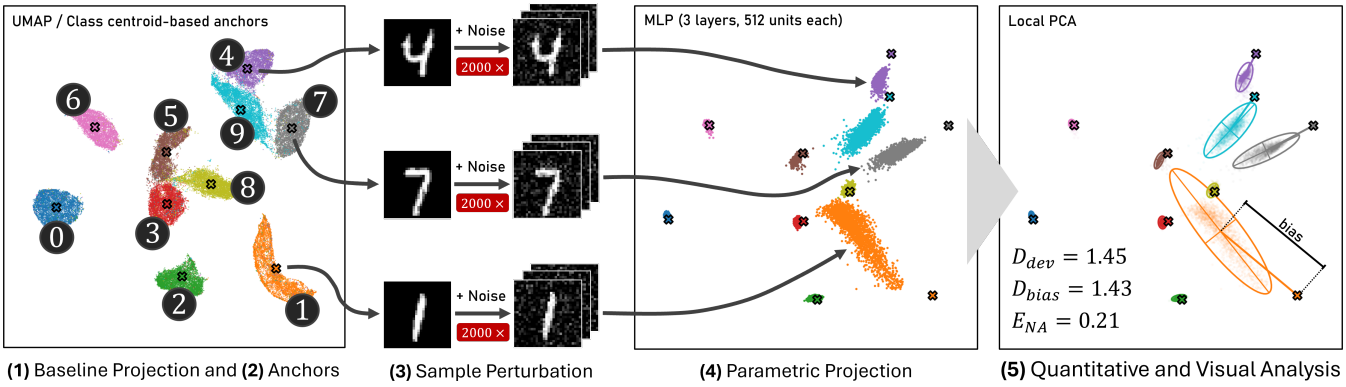


Figure 1: *Framework overview on MNIST. **(1)** A baseline UMAP projection is computed. **(2)** Class centroid-based anchors are selected. **(3)** Isotropic Gaussian noise is added to perturb anchor samples. **(4)** A multi-layer perceptron (MLP) projects all noisy samples into 2D. **(5)* Stability is assessed with three metrics: Mean displacement (D_{dev}), displacement bias (D_{bias}), and nearest-anchor assignment error (E_{NA}); lower is better. Here, we show a local PCA visualization of displacement bias and the amplification or attenuation along principal directions.*

Abstract

Parametric projections let analysts embed new points in real time, but input variations from measurement noise or data drift can produce unpredictable shifts in the 2D layout. Whether and where a projection is locally stable remains largely unexamined. In this paper, we present a stability evaluation framework that probes parametric projections with Gaussian perturbations around selected anchor points and assesses how neighborhoods deform in the 2D embedding. Our approach combines quantitative measures of mean displacement, bias, and nearest-anchor assignment error with per-anchor visualizations of displacement vectors, local PCA ellipsoids, and Voronoi misassignment for detailed inspection. We demonstrate the framework's effectiveness on UMAP- and t-SNE-based neural projectors of varying network sizes and study the effect of Jacobian regularization as a gradient-based robustness strategy. We apply our framework to the MNIST and Fashion-MNIST datasets. The results show that our framework identifies unstable projection regions invisible to reconstruction error or neighborhood-preservation metrics.

CCS Concepts

• ***Human-centered computing*** → ***Visualization;*** • ***Computing methodologies*** → *Machine learning;*

1. Introduction

Making sense of high-dimensional data requires reducing it to a form humans can interpret. Dimensionality reduction (DR) methods address this by mapping data to two or three dimensions while preserving relationships such as distances and neighborhood structures [NA19]. A common shortcoming of these approaches is that they cannot represent the projection as an explicit parametric function, preventing consistent and efficient projection of new or synthetic data without recomputing the entire embedding [vdM09]. In recent years, neural networks (NNs) have increasingly been used to learn such parametric mappings [EHT20]. These approaches can approximate non-linear projections and can generalize to a variety of DR techniques [DGB*25]. With NNs, once trained, the mapping can be evaluated in constant time per point, enabling fast out-of-sample

DOI: 10.2312/eurova.20261015

projection of new or synthetic data without recomputing global pairwise relationships [SMG21]. In contrast, classical DR techniques such as *UMAP* [MHSG18] and *t-SNE* [vdMH08] are non-parametric and require full recomputation when new data arrives. This makes feed-forward multi-layer perceptron (MLP) projections attractive for *streaming* and *interactive settings*, for deployment scenarios where embeddings must be produced on demand, and for workflows that require a consistent layout across updates. For visual analytics, this *consistency* is essential, since analysts must be able to *trust* that spatial patterns reflect data structures rather than artifacts of recomputation [NDKS22]. However, parametric projections introduce a concern: The learned mapping is constrained only by the training data and objective, and its behavior under small input perturbations is rarely examined [DGB*25]. Current evaluation methods emphasize accurate local neighborhood preservation [VK01], but ignore sensitivity to inputs that are slightly perturbed. In this work, we analyze MLP projection architectures and how well they preserve local neighborhoods under input variations (see Fig. 1). Our evaluation combines *quantitative measures* with *visual analysis* to reveal systematic and sometimes surprising differences in how MLP-based parametric projections respond to small Gaussian perturbations. In this paper, we contribute:

(1) *Three quantitative measures* to assess *local neighborhood stability* in parametric projections under small input perturbations.
(2) *Three visualizations* that link our local stability measures to spatial and neighborhood changes in the projection.
(3) *Empirical comparisons* of parametric projections for UMAP and t-SNE, validating the effectiveness of stability assessment.
(4) We share the analysis, results, and source code on OSF and GitHub for *reproducibility*.

2. Related Work

Projection Methods for Visualization: We define a high-dimensional dataset as $D = \{x_i\}_{1 \le i \le n}$ with n samples $x_i \in \mathbb{R}^d$ and a *projection* method P that maps D to a lower-dimensional representation as $P(D) = \{P(x_i) \mid x_i \in D\} = \{y_i\}_{1 \le i \le n}$, where $P(D) \subset \mathbb{R}^q$ with $q \ll d$. In our setting, $q = 2$, allowing $P(D)$ to be visualized as a two-dimensional scatterplot. Projection methods have been extensively studied and evaluated in several surveys [CG15, EMK*19, NA19]. They are commonly categorized as either *linear* or *non-linear* methods [LMW*16, EMK*19], and further distinguished by whether they primarily preserve *global* structure or *local* neighborhood relations. Linear projection techniques such as PCA can be computed very efficiently and are known to preserve global variance structure, while MDS [KW78] provides a non-linear alternative with a similar global emphasis. In contrast, many non-linear methods prioritize the preservation of local neighborhoods at the expense of global structure, including t-SNE [vdMH08] and UMAP [MHSG18]. Among other projection quality measures [EMK*19], Venna and Kaski [VK01] proposed *Trustworthiness* and *Continuity*, measuring intrusions and extrusions in local neighborhoods.

Parametric Projections: Standard non-linear DR methods [KW78, vdMH08, MHSG18] are non-parametric and typically require recomputation when projecting new data points [HHKS23]. Parametric approaches address this limitation by learning explicit mapping functions, often implemented using neural networks (NNs), from the input space to a lower-dimensional embedding [BBH12]. Van der Maaten [vdM09] introduced *parametric t-SNE* using a feed-forward NN to approximate the non-parametric embedding. Similarly, *parametric UMAP* [SMG21] employs NNs, including autoencoder-based architectures, to replace the non-parametric optimization step. By training NNs to directly infer low-dimensional coordinates, Espadoto et al. [EHT20] demonstrated that sufficiently expressive models can approximate a wide range of existing non-parametric DR techniques. *HyperNP* [AEC*22] extends this idea by learning parametric approximations across DR hyperparameters (e.g., perplexity in t-SNE), enabling interactive exploration. Finally, *ParaDime* [HHKS23] formalizes the design of NN-based DR methods through a grammar for parametric DR. Recent approaches use autoencoders to learn parametric mappings [DGB*25].

Robustness and Adversarial Perturbations: Adversarial robustness concerns carefully crafted input perturbations designed to cause maximal change in model output under minimal input changes (i.e., small-norm constraint) [GSS15]. Recent work has highlighted that parametric DR models can be vulnerable to attacks [FKW*25]. Adversarial robustness is a training objective [MMS*18]. Cohen et al. [CRK19] show that classifiers robust to Gaussian input noise are certifiably robust to adversarial perturbations. Fast adversarial training [WRK20] provides a baseline defense mechanism, while Kabaha and Drachsler-Cohen [KDC24] introduced verification methods for computing global robustness bounds of neural networks. Lin and Fukuyama [LF24] developed a framework for calibrating DR hyperparameters in the presence of noise, addressing overfitting issues in t-SNE and UMAP. Beyond adversarial training, explicit architectural constraints can enforce smoothness. Jacobian regularization [JG18] penalizes the Frobenius norm of the output Jacobian with respect to inputs. Spectral normalization [MKKY18] bounds network Lipschitz constants by constraining the spectral norm of weight matrices.

Visual Analysis of Point Distributions: DR methods produce 2D scatterplots with visualization challenges, specifically in the context of representing clusters and overplotting. Instead of rendering individual points, density plots color-encode aggregated point counts, typically smoothed via kernel density estimation [Sil86, Sco92]. Ellis and Dix [ED07] provide a taxonomy of clutter reduction techniques including filtering, sampling, opacity adjustment, and spatial distortion. Von Landesberger et al. [vLBRS09] summarize point distributions using geometric primitives including convex hulls and minimum spanning trees. Chen et al. [CCM*14] explored hierarchical multi-scale sampling that maintains relative densities and outliers. Introducing visual abstractions, Liao et al. [LWCC18] replace dense point groups with aggregate visual marks such as ellipses or convex hulls, reducing clutter while conveying cluster shape.

3. Stability Evaluation Framework

Our framework evaluates local stability by measuring how a given MLP projection responds to small input perturbations around representative *anchor points*, essentially answering the question: *How stable is it under plausible measurement noise?* Given a projection $f : \mathbb{R}^d \to \mathbb{R}^q$, a high-dimensional dataset, and a strategy to select anchor points: (1) We fit a baseline projection method (e.g., UMAP or t-SNE) to obtain embedded coordinates. (2) We choose a set of anchor points based on the chosen strategy. When class labels are available (as in our datasets), selecting samples near class

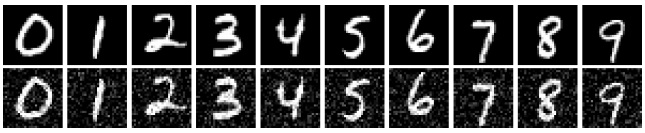


Figure 2: *Top – Class centroid-based anchors of MNIST. Bottom – Same images with isotropic Gaussian noise added ($\sigma = 0.17$).*

centroids in projection space ensures representative anchors rather than outliers. (3) We generate perturbed inputs around each anchor by adding isotropic Gaussian noise. (4) We project the perturbed samples and compute quantitative stability metrics. (5) We visualize the resulting point clouds to reveal local geometric structure. This framework applies uniformly to MLP-based parametric projections.

Local Neighborhood Stability: For each anchor point x_0, we probe local stability by adding isotropic Gaussian noise parameterized by a target input-space radius $r > 0$. Concretely, let $\varepsilon \sim \mathcal{N}(0, \sigma^2 I_d)$ in d dimensions; then

$$r := \sqrt{\mathbb{E}\left[\|\varepsilon\|_2^2\right]} = \sigma\sqrt{d}, \text{ with } \sigma = \frac{r}{\sqrt{d}}, \text{ and } \mathbb{E}\left[\|\varepsilon\|_2^2\right] = d\sigma^2, \quad (1)$$

so that specifying a radius r uniquely determines the noise scale. In practice, r is chosen to represent a small, interpretable neighborhood around each data point (e.g., such that perturbed samples remain visually recognizable for image data). As a heuristic, r may be selected as a small fraction of a typical input-space distance (i.e., a very low percentile of all pairwise distances). We choose the 0.25th percentile. For image data bounded to $[0,1]$, we clip perturbed values element-wise to this range (Fig. 2). We generate N perturbed samples per anchor and clip them to obtain $\tilde{x}_i = \text{clip}(x_0 + \varepsilon_i)$. Clipping reduces the realized perturbation magnitude relative to the nominal radius r. We report the effective (root-mean-square) radius $r_{\text{eff}} := \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|\tilde{x}_i - x_0\|_2^2}$, computed from the clipped perturbations, the target radius r, and the resulting σ in Tab. 1. The ratio r_{eff}/r quantifies the clipping effect. Values near 1 indicate negligible clipping, while lower ratios reflect boundary saturation.

Dataset	n	d	r	r_{eff}	r_{eff}/r	σ
MNIST [LCB98]	70000	784	4.73	3.46	0.73	0.17
Fashion [XRV17]	70000	784	4.47	3.69	0.83	0.16

Table 1: *Size* **n**, *dimensionality* **d**, *noise radius* **r**, *effective noise radius* $\mathbf{r}_{\text{eff}}$, *the ratio* $\mathbf{r}_{\text{eff}}/\mathbf{r}$, *std. dev.* σ. *Noisy samples are created with the 0.25th pairwise-distance percentile, 5 anchors/class, and N=1000 per anchor, using more samples for* $\mathbf{r}_{\text{eff}}$ *precision (Sec. 4).*

Stability Measures: A stable projection maps similar inputs to similar outputs. Small perturbations in input space should yield small displacements in projection space, while unstable projections exhibit large, erratic, or systematically biased responses to minor input changes. Thus, we define a clean anchor point x_0 with projected location $z_0 = f(x_0)$, and the projected perturbed samples $z_i = f(\tilde{x}_i)$ for $i = 1, \ldots, N$. We propose three quantitative measures:

(1) Mean Displacement: The typical noise-induced drift at noise level σ is quantified as

$$D_{\text{dev}}(\sigma) := \frac{1}{N}\sum_{i=1}^{N} \|z_i - z_0\|_2 \approx \mathbb{E}[\|z - z_0\|], \quad (2)$$

where N is the number of noisy samples.

(2) Displacement Bias: Systematic displacement of the mean projection from the anchor is measured as

$$D_{\text{bias}}(\sigma) := \left\| \frac{1}{N}\sum_{i=1}^{N} z_i - z_0 \right\|_2 \approx \|\mathbb{E}[z] - z_0\| \quad (3)$$

(3) Nearest-Anchor Assignment Error: For each anchor $z_0^{(a)}$ and its corresponding noise-perturbed projections $\{z_i^{(a)}\}_{i=1}^{N}$, we assign each projected point to its nearest anchor in the projected space:

$$\hat{a}(z) = \arg\min_k \|z - z_0^{(k)}\|_2. \quad (4)$$

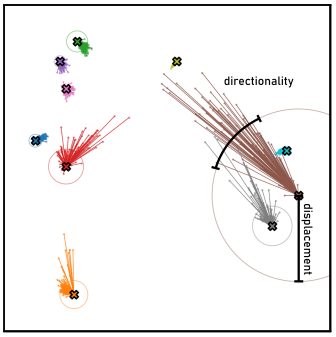
We then define the assignment error at noise level σ as

$$E_{\text{NA}}(\sigma) := \frac{1}{A}\sum_{a=1}^{A} \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\left[\hat{a}\left(z_i^{(a)}\right) \neq a\right], \quad (5)$$

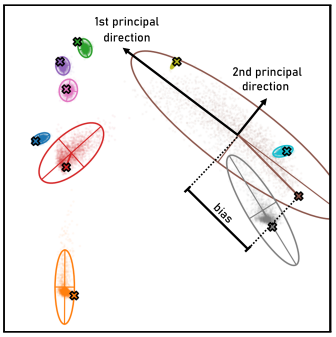
where A denotes the number of anchors and $\mathbf{1}[\cdot]$ is the indicator function. This measure quantifies the probability that noise-induced perturbations cause a sample to leave the Voronoi region of its original anchor, providing a direct notion of projection robustness.

Lower values of D_{dev}, D_{bias}, and E_{NA} indicate greater stability. D_{dev} measures typical displacement magnitude; D_{bias} detects systematic drift in a consistent direction. In our experiments, we use $N = 2000$ samples for computing D_{dev}, D_{bias}, and E_{NA}.

Visual Diagnostics: We propose three visualizations to assess the local neighborhood stability of parametric projections. In all visualizations, anchor points are visualized as crosses.

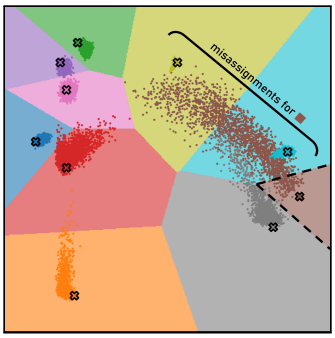


Anchor Lines: Anchor lines are geometric primitives [vLBRS09] and explicitly encode the *relational* structure between each noisy projection and its corresponding anchor. By visualizing displacement vectors from the anchor to the noisy samples, this representation highlights directional bias, coherence, and asymmetry in the projected perturbations. It enables assessment of whether noise induces systematic drift away from the anchor or remains approximately isotropic. The *mean displacement* (D_{dev}) of the anchor is represented as a circle around the anchor point and captures the overall instability.



Local PCA: Local principal component analysis (PCA) captures the *differential* structure of the projection around the anchor by approximating the local geometry of the noise-induced point cloud. The resulting principal directions and variances indicate how perturbations are amplified or attenuated along different directions in the projected

space. This approach is inspired by Liao et al. [LWCC18]. Changes in orientation or anisotropy provide evidence of deviations from locally linear noise propagation. The local PCA ellipsoid is centered at the mean of the noise-induced projections and thus visually encodes the anchor's *displacement bias* (D_{bias}). As an alternative, we propose Density Contours in the supplementary material.



Voronoi Assignment: This visualization introduces a *comparative* perspective when multiple anchors are present. By partitioning the projection space according to the nearest anchor, Voronoi regions [Aur91] indicate which reference point dominates locally. Points from one anchor's perturbation cloud that fall into a neighboring anchor's Voronoi cell signal potential misassignment under noise, identifying areas of reduced robustness with respect to competing anchors, visually linked to the *nearest-anchor assignment error* (E_{NA}).

4. Evaluation

Experimental Setup: We use the two datasets in Tab. 1 and evaluate both UMAP and t-SNE as base projection methods [BWT*24]. We split each dataset using an 80-10-10 train-validation-test protocol. Quality and stability metrics are computed on the held-out test set and averaged across 10 runs with different seeds (see Tab. 2). Each run refits the base projection using the specified seed and uses independent network initializations as well as noise draws; anchors are class-centroid-based with count equal to the number of classes. We report *Trustworthiness* and *Continuity* measures [VK01] as an average of all $T(k)$ with neighborhood size $k \in \{2,4,8,\ldots,n/2\}$ and $C(k)$ respectively [CPA*20]. Higher T and C indicate better local structure preservation; results are in Tab. 2.

Neural Network Architectures and Training: We train fully connected MLPs with ReLU activations learning $f : \mathbb{R}^d \to \mathbb{R}^2$. `MLP-small`: 3 hidden layers, 512 units each; `MLP-large`: 6 hidden layers, 1024 units each. Our general objective function is $\mathcal{L}_{\text{MSE}} = \mathbb{E}_{(x,y)}\left[\|f(x)-y\|_2^2\right]$ [EHT20, DGB*25]. We use the Adam optimizer [KB15], batch size 256, learning rate 10^{-3}, and 100 epochs with early stopping (patience=10). No noise augmentation is applied during training. The `+J` suffix denotes the application of Jacobian regularization. It penalizes the Frobenius norm of the input-output Jacobian $J_f(x) = \partial f_\theta(x)/\partial x \in \mathbb{R}^{k\times d}$, adding $\mathcal{L}_{\text{Jac}} = \lambda \mathbb{E}_x\left[\|J_f(x)\|_F^2\right]$ to the training loss (we choose λ=10; see sweep over λ in supplementary material), encouraging locally smooth mappings that are more robust to small input perturbations [JG18].

Quantitative Results: Tab. 2 summarizes six metrics across two datasets and two projection methods (UMAP, t-SNE), averaged over 10 runs. Jacobian regularization consistently improves stability for both architectures and both projection methods: `MLP-large+J` achieves the lowest mean displacement in all settings, reducing displacement by 28–63% relative to its unregularized counterpart `MLP-large`. The effect is equally pronounced for the smaller network:

Model	MNIST (UMAP)	Fashion (UMAP)	MNIST (t-SNE)	Fashion (t-SNE)
Average MSE Loss (lower is better)				
`MLP-small`	.829 ± .104	.293 ± .054	62.7 ± 3.8	30.7 ± 2.98
`MLP-small+J`	.676 ± .035	**.279 ± .046**	61.8 ± 2.15	29.9 ± 2.81
`MLP-large`	.855 ± .108	.33 ± .044	52.7 ± 6.34	32 ± 3.34
`MLP-large+J`	**.668 ± .06**	.297 ± .049	**48.8 ± 4.9**	**26.1 ± 1.71**
Avg. Trustworthiness $T(k)$ with $k \in \{2,4,8,\ldots,n/2\}$ (higher is better)				
`MLP-small`	.859 ± .004	.928 ± .002	.857 ± .001	.928 ± .001
`MLP-small+J`	.858 ± .002	.927 ± .002	.85 ± .001	.926 ± .001
`MLP-large`	**.867 ± .003**	**.929 ± .002**	**.872 ± .001**	**.931 ± .001**
`MLP-large+J`	.865 ± .001	.929 ± .002	.863 ± .001	.929 ± .001
Avg. Continuity $C(k)$ with $k \in \{2,4,8,\ldots,n/2\}$ (higher is better)				
`MLP-small`	.864 ± .001	.938 ± .002	.871 ± .001	.943 ± .001
`MLP-small+J`	**.866 ± .001**	**.939 ± .002**	**.873 ± .001**	**.944 ± .001**
`MLP-large`	.864 ± .002	.938 ± .002	.871 ± .001	.942 ± .001
`MLP-large+J`	.866 ± .001	.939 ± .002	.872 ± .001	.943 ± .001
Mean Displacement (lower is better)				
`MLP-small`	1.8 ± .382	.863 ± .175	24.8 ± 1.59	11.7 ± 2.69
`MLP-small+J`	.265 ± .051	.465 ± .204	5.03 ± .612	6.06 ± 1.38
`MLP-large`	.562 ± .282	.665 ± .244	6.34 ± 1.41	10.7 ± 2.55
`MLP-large+J`	**.207 ± .069**	**.391 ± .091**	**4.58 ± 1.4**	**5.35 ± 1.36**
Displacement Bias (lower is better)				
`MLP-small`	1.76 ± .375	.772 ± .179	24.4 ± 1.62	10.4 ± 2.74
`MLP-small+J`	.233 ± .053	.435 ± .208	4.53 ± .617	5.47 ± 1.37
`MLP-large`	.524 ± .282	.622 ± .246	5.35 ± 1.63	9.67 ± 2.65
`MLP-large+J`	**.18 ± .066**	**.361 ± .093**	**4.13 ± 1.41**	**4.78 ± 1.34**
Average Nearest-Anchor Assignment Error (lower is better)				
`MLP-small`	.294 ± .069	.109 ± .038	.307 ± .075	.192 ± .081
`MLP-small+J`	.006 ± .012	.037 ± .039	**.017 ± .022**	.068 ± .053
`MLP-large`	.069 ± .06	.078 ± .047	.024 ± .022	.164 ± .091
`MLP-large+J`	**.003 ± .009**	**.031 ± .034**	.019 ± .031	**.036 ± .046**

Table 2: *Stability and quality metrics for MLP-based parametric projections based on UMAP and t-SNE with different regularization strategies across datasets. Bold values indicate best, underlined values worst performance per metric and dataset.*

on *MNIST* (UMAP), `MLP-small+J` reduces mean displacement from 1.80 to .265 (−85%) and anchor assignment error from .294 to .006. The same pattern holds for t-SNE: on *MNIST* (t-SNE), `MLP-small+J` reduces displacement from 24.8 to 5.03 (−80%) and anchor error from .307 to .017. Notably, on *MNIST* (UMAP), the unregularized `MLP-small` exhibits mean displacement (1.80) nearly equal to displacement bias (1.76), indicating systematic drift rather than isotropic spread; the same effect appears on *MNIST* (t-SNE) with displacement 24.8 vs. bias 24.4. Increasing network capacity alone provides moderate stability gains (`MLP-large` vs. `MLP-small`), but Jacobian regularization is substantially more effective: On *Fashion* (UMAP), `MLP-large`'s displacement drops to .665, whereas adding `+J` further lowers it to .391. Reconstruction loss and stability do not trade off uniformly; `MLP-large+J` achieves the best MSE on *MNIST* (UMAP) (.668) while simultaneously being the most stable. The averaged *Trustworthiness* and *Continuity* remain near-constant across all methods (T: .850–.931, C: .864–.944), confirming that neighborhood-preservation metrics fail to capture the stability differences revealed by displacement-based metrics, i.e., the failure mode anticipated in Sec. 1. Our stability metrics are cheap to compute. The total wall-clock per model on

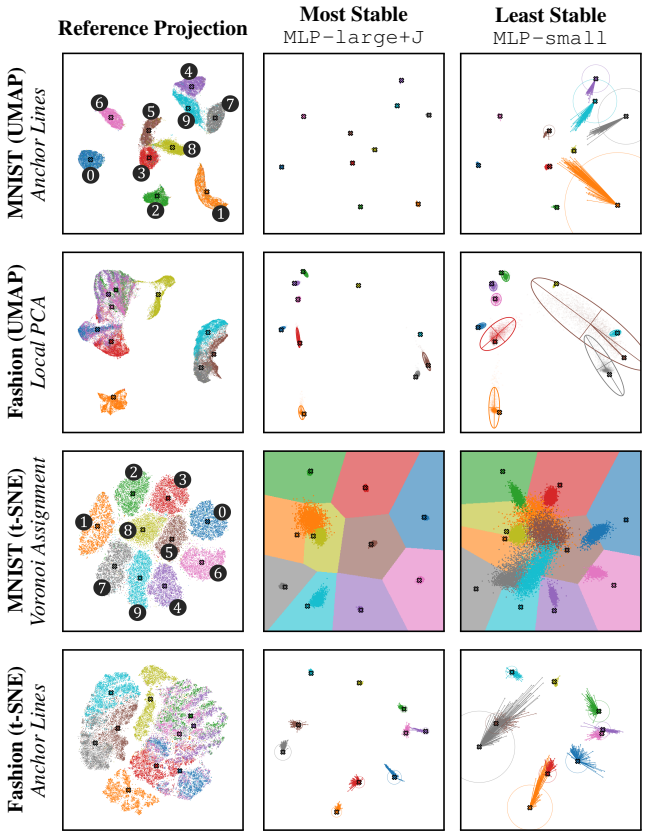


Figure 3: *Qualitative stability comparison across UMAP and t-SNE projections. Each row shows the reference projection with anchors, the most and least stable MLP by mean displacement (Tab. 2) with stability visualizations.*

MNIST/*Fashion* is ∼135 ms for `MLP-large`, dominated by MLP forward passes; E_{NA} is $O(NA^2)$ and sub-millisecond.

Qualitative Results: Fig. 3 contrasts the most (`MLP-large+J`) and least (`MLP-small`) stable models per setting by mean displacement, each row using a different stability visualization alongside the reference projection. On *MNIST* (UMAP), anchor lines for `MLP-large+J` remain tightly clustered around anchors, while `MLP-small` shows wide displacement fans. For *Fashion* (UMAP), local PCA ellipses reveal both larger extent and stronger directional anisotropy for `MLP-small`, consistent with its higher displacement bias. The *MNIST* (t-SNE) Voronoi row provides the starkest contrast: `MLP-large+J` confines perturbed points within their anchor's Voronoi cell (nearest-anchor assignment error .078), while `MLP-small` scatters points across cell boundaries (.329), with displacement nearly entirely due to systematic bias. On *Fashion* (t-SNE), shorter anchor lines confirm reduced displacement for `MLP-large+J` relative to `MLP-small`.

5. Discussion

Our evaluation yields two main findings: (1) Standard neighborhood-based quality metrics (Trustworthiness, Continuity) remain virtually identical across all tested methods (T: .850–.931, C: .864–.944), failing to reflect substantial stability differences. Our displacement-based measures and visual diagnostics close this gap, revealing instabilities that T and C cannot capture. (2) As a practical demonstration, Jacobian regularization, with an expected but now quantifiable effect, reduces displacement by up to 85% (UMAP) and 80% (t-SNE), and anchor assignment error to near zero on *MNIST*, with consistent gains across both projection methods and datasets. Comparing two network sizes with and without regularization confirms that Jacobian regularization is more effective than increasing capacity alone. We also test spectral normalization and report results on two additional datasets in the supplementary material.

Implications for Visual Analytics: The stability of MLP projections directly determines whether analysts can trust that spatial patterns represent genuine data structures [NA19]. Our evaluation shows that standard neighborhood-preservation metrics are insufficient: Methods with near-identical Trustworthiness and Continuity scores exhibit displacement differences exceeding 80%. Our visual diagnostics let practitioners inspect *where* and *in which direction* a projection is sensitive. Our findings indicate that regularization strategy matters more than network capacity: A compact Jacobian-regularized network achieves stability at lower computational cost (training times are reported in the supplementary material). Overall, we advocate for incorporating stability assessment when training MLP-based parametric projections.

Limitations and Future Work: Jacobian regularization minimizes local sensitivity; displacement measures quantify local sensitivity. Hence, the stability improvement is expected. We analyze the cost-benefit tradeoff (stability gain vs. MSE/T/C cost) rather than an unexpected effect. Our measurements characterize the learned parametric mapping, not the base DR, and do not disentangle NN- from DR-induced effects. Isotropic Gaussian noise may not match realistic perturbation distributions for image data, though in our study it serves as a proxy for sensor drift; more realistic perturbations could be obtained by interpolating between same-class samples or via inverse projection [EAS*21, DGB*25]. Our measures are defined as functions of σ, but we evaluate only a single noise level per dataset; a sweep across σ values would provide a more complete stability profile. We select σ in a dataset-adaptive manner via the 0.25th percentile of pairwise distances (Tab. 1), ensuring perturbations remain within a realistic range for each dataset. We evaluate only MLP architectures; extending the framework to AEs, CNNs, or other parametric DR variants would broaden its applicability. Only class-centroid-based anchors are tested; thus, an analysis of other anchor selection strategies would strengthen the evaluation.

6. Conclusion

Parametric projections promise real-time inference and out-of-sample extension, yet standard quality metrics ignore their sensitivity to input perturbations. We introduced displacement-based stability measures and visualizations that reveal instabilities invisible to *Trustworthiness* and *Continuity* measures. Jacobian regularization can mitigate such instability, as confirmed by our diagnostic measures, at minimal cost to reconstruction fidelity. The takeaway is concrete: Pair parametric projections with smoothness constraints and validate stability before deployment in noise-prone scenarios.

Acknowledgments: This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 251654672 – TRR 161 (Project A03).

References

[AEC*22] APPLEBY G., ESPADOTO M., CHEN R., GOREE S., TELEA A. C., ANDERSON E. W., CHANG R.: HyperNP: Interactive visual exploration of multidimensional projection hyperparameters. *Comput. Graph. Forum 41*, 3 (2022), 169–181. doi:10.1111/CGF.14531. 2

[Aur91] AURENHAMMER F.: Voronoi diagrams - A survey of a fundamental geometric data structure. *ACM Comput. Surv. 23*, 3 (1991), 345–405. doi:10.1145/116873.116880. 4

[BBH12] BUNTE K., BIEHL M., HAMMER B.: A general framework for dimensionality-reducing data visualization mapping. *Neural Comput. 24*, 3 (2012), 771–804. doi:10.1162/NECO_A_00250. 2

[BWT*24] BLUMBERG D., WANG Y., TELEA A., KEIM D. A., DENNIG F. L.: Inverting Multidimensional Scaling Projections Using Data Point Multilateration. In *15th Int. EuroVis Workshop Vis. Anal.* (2024). doi:10.2312/eurova.20241112. 4

[CCM*14] CHEN H., CHEN W., MEI H., LIU Z., ZHOU K., CHEN W., GU W., MA K.: Visual abstraction and exploration of multi-class scatterplots. *IEEE Trans. Vis. Comput. Graph. 20*, 12 (2014), 1683–1692. doi:10.1109/TVCG.2014.2346594. 2

[CG15] CUNNINGHAM J. P., GHAHRAMANI Z.: Linear dimensionality reduction: Survey, insights, and generalizations. *J. Mach. Learn. Res. 16* (2015), 2859–2900. doi:10.5555/2789272.2912091. 2

[CPA*20] COLANGE B., PELTONEN J., AUPETIT M., DUTYKH D., LESPINATS S.: Steering distortions to preserve classes and neighbors in supervised dimensionality reduction. In *Adv. Neural Inf. Process. Syst.* (2020), vol. 33, pp. 13214–13225. 4

[CRK19] COHEN J., ROSENFELD E., KOLTER J. Z.: Certified adversarial robustness via randomized smoothing. In *36th Int. Conf. Mach. Learn.* (2019), pp. 1310–1320. URL: http://proceedings.mlr.press/v97/cohen19c.html. 2

[DGB*25] DENNIG F. L., GEYER N., BLUMBERG D., METZ Y., KEIM D. A.: Evaluating Autoencoders for Parametric and Invertible Multidimensional Projections. In *16th Int. EuroVis Workshop Vis. Anal.* (2025). doi:10.2312/eurova.20251099. 1, 2, 4, 5

[EAS*21] ESPADOTO M., APPLEBY G., SUH A., CASHMAN D., LI M., SCHEIDEGGER C., ANDERSON E. W., ANDD ALEXANDRU C. TELEA R. C.: UnProjection: Leveraging inverse-projections for visual analytics of high-dimensional data. *IEEE Trans. Vis. Comput. Graph. 29*, 2 (2021), 1559–1572. doi:10.1109/TVCG.2021.3125576. 5

[ED07] ELLIS G., DIX A.: A taxonomy of clutter reduction for information visualisation. *IEEE Trans. Vis. Comput. Graph. 13*, 6 (2007), 1216–1223. doi:10.1109/TVCG.2007.70535. 2

[EHT20] ESPADOTO M., HIRATA N. S. T., TELEA A. C.: Deep learning multidimensional projections. *Inf. Vis. 19*, 3 (2020), 247–269. doi:10.1177/1473871620909485. 1, 2, 4

[EMK*19] ESPADOTO M., MARTINS R. M., KERREN A., HIRATA N. S. T., TELEA A. C.: Toward a quantitative survey of dimension reduction techniques. *IEEE Trans. Vis. Comput. Graph. 27*, 3 (2019), 2153–2173. doi:10.1109/TVCG.2019.2944182. 2

[FKW*25] FUJIWARA T., KUCHER K., WANG J., MARTINS R. M., KERREN A., YNNERMAN A.: Adversarial attacks on machine learning-aided visualizations. *J. Vis. 28*, 1 (2025), 133–151. doi:10.1007/S12650-024-01029-2. 2

[GSS15] GOODFELLOW I. J., SHLENS J., SZEGEDY C.: Explaining and harnessing adversarial examples. In *3rd Int. Conf. Learn. Represent.* (2015). URL: http://arxiv.org/abs/1412.6572. 2

[HHKS23] HINTERREITER A. P., HUMER C., KAINZ B., STREIT M.: ParaDime: A framework for parametric dimensionality reduction. *Comput. Graph. Forum 42*, 3 (2023), 337–348. doi:10.1111/CGF.14834. 2

[JG18] JAKUBOVITZ D., GIRYES R.: Improving DNN robustness to adversarial attacks using jacobian regularization. In *Comput. Vis.* (2018), vol. 11216, pp. 525–541. doi:10.1007/978-3-030-01258-8_32. 2, 4

[KB15] KINGMA D. P., BA J.: Adam: A method for stochastic optimization. In *3rd Int. Conf. Learn. Represent.* (2015). 4

[KDC24] KABAHA A., DRACHSLER-COHEN D.: Verification of neural networks' global robustness. *ACM Program. Lang. 8*, OOPSLA1 (2024), 1010–1039. doi:10.1145/3649847. 2

[KW78] KRUSKAL J., WISH M.: Multidimensional scaling. *Murry Hill* (1978). doi:10.4135/9781412985130. 2

[LCB98] LECUN Y., CORTES C., BURGES C.: MNIST handwritten digit database, 1998. 3

[LF24] LIN J., FUKUYAMA J.: Calibrating dimension reduction hyperparameters in the presence of noise. *PLoS Comput. Biol. 20*, 9 (2024), e1012427. doi:10.1371/journal.pcbi.1012427. 2

[LMW*16] LIU S., MALJOVEC D., WANG B., BREMER P., PASCUCCI V.: Visualizing high-dimensional data: Advances in the past decade. *IEEE Trans. Vis. Comput. Graph. 23*, 3 (2016), 1249–1268. doi:10.1109/TVCG.2016.2640960. 2

[LWCC18] LIAO H., WU Y., CHEN L., CHEN W.: Cluster-based visual abstraction for multivariate scatterplots. *IEEE Trans. Vis. Comput. Graph. 24*, 9 (2018), 2531–2545. doi:10.1109/TVCG.2017.2754480. 2, 4

[MHSG18] MCINNES L., HEALY J., SAUL N., GROSSBERGER L.: UMAP: Uniform Manifold Approximation and Projection. *J. Open Source Softw. 3*, 29 (2018), 861. doi:10.21105/joss.00861. 2

[MKKY18] MIYATO T., KATAOKA T., KOYAMA M., YOSHIDA Y.: Spectral normalization for generative adversarial networks. In *6th Int. Conf. Learn. Represent.* (2018). 2

[MMS*18] MADRY A., MAKELOV A., SCHMIDT L., TSIPRAS D., VLADU A.: Towards deep learning models resistant to adversarial attacks. In *6th Int. Conf. Learn. Represent.* (2018). 2

[NA19] NONATO L. G., AUPETIT M.: Multidimensional projection for visual analytics: Linking techniques with distortions, tasks, and layout enrichment. *IEEE Trans. Vis. Comput. Graph. 25*, 8 (2019). doi:10.1109/TVCG.2018.2846735. 1, 2, 5

[NDKS22] NGO Q. Q., DENNIG F. L., KEIM D. A., SEDLMAIR M.: Machine learning meets visualization - Experiences and lessons learned. *it - Information Technology 64*, 4–5 (2022), 169–180. doi:10.1515/ITIT-2022-0034. 2

[Sco92] SCOTT D. W.: *Multivariate Density Estimation: Theory, Practice, and Visualization*. Wiley, 1992. doi:10.1002/9780470316849. 2

[Sil86] SILVERMAN B. W.: *Density Estimation for Statistics and Data Analysis*. Springer, 1986. doi:10.1007/978-1-4899-3324-9. 2

[SMG21] SAINBURG T., MCINNES L., GENTNER T. Q.: Parametric UMAP embeddings for representation and semisupervised learning. *Neural Comput. 33*, 11 (2021), 2881–2907. doi:10.1162/NECO_A_01434. 2

[vdM09] VAN DER MAATEN L.: Learning a parametric embedding by preserving local structure. In *12th Int. Conf. Artif. Intell. Stat.* (2009), vol. 5, pp. 384–391. 1, 2

[vdMH08] VAN DER MAATEN L., HINTON G.: Visualizing data using t-SNE. *J. Mach. Learn. Res. 9*, 86 (2008), 2579–2605. 2

[VK01] VENNA J., KASKI S.: Neighborhood preservation in nonlinear projection methods: An experimental study. In *30th Int. Conf. Artif. Neural Netw.* (2001), vol. 2130, pp. 485–491. doi:10.1007/3-540-44668-0_68. 2, 4

[vLBRS09] VON LANDESBERGER T., BREMM S., REZAEI M., SCHRECK T.: Visual analytics of time dependent 2D point clouds. In *Comput. Graph. Int.* (2009), pp. 97–101. doi:10.1145/1629739.1629751. 2, 3

[WRK20] WONG E., RICE L., KOLTER J. Z.: Fast is better than free: Revisiting adversarial training. In *8th Int. Conf. Learn. Represent.* (2020). URL: https://openreview.net/forum?id=BJx040EFvH. 2

[XRV17] XIAO H., RASUL K., VOLLGRAF R.: Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *CoRR abs/1708.07747* (2017). arXiv:1708.07747. 3