A Connection Between Score Matching and Local Intrinsic Dimension

Eric Yeats
PNNL

Aaron Jacobson[†]
UNC Chapel Hill

Darryl Hannan
PNNL

Yiran Jia[†]
UC San Diego

Timothy Doster
PNNL

Henry Kvinge
PNNL

Scott Mahan
PNNL

`{first}.{last}@pnnl.gov`

Abstract

The local intrinsic dimension (LID) of data is a fundamental quantity in signal processing and learning theory, but quantifying the LID of high-dimensional, complex data has been a historically challenging task. Recent works have discovered that diffusion models capture the LID of data through the spectra of their score estimates and through the rate of change of their density estimates under various noise perturbations. While these methods can accurately quantify LID, they require either many forward passes of the diffusion model or use of gradient computation, limiting their applicability in compute- and memory-constrained scenarios.

We show that the LID is a lower bound on the denoising score matching loss, motivating use of the denoising score matching loss as a LID estimator. Moreover, we show that the equivalent implicit score matching loss also approximates LID via the normal dimension and is closely related to a recent LID estimator, FLIPD. Our experiments on a manifold benchmark and with Stable Diffusion 3.5 indicate that the denoising score matching loss is a highly competitive and scalable LID estimator, achieving superior accuracy and memory footprint under increasing problem size and quantization level.

1 Introduction

Observations of high-dimensional data which are generated by physics or other natural phenomena tend to inherit low-dimensional structure. This is commonly referred to as the manifold hypothesis, and it underpins central assumptions in machine learning [2]. The fundamental quantity encapsulating the lower dimensional structure of data is the local intrinsic dimension (LID). For a point x on a data manifold, the LID is the local number of dimensions required to losslessly encode the data around x.

The LID has clear implications in signal processing, as it determines bounds on how (locally) compressible a distribution is [4]. Moreover, the LID is vital to deep learning [14], where learning from high-dimensional data is made possible by its relatively low-dimensional structure. More specifically, lower LID improves the statistical efficiency of learning - lower dimensional structure makes learning and generalization easier [19]. The LID is also a practical tool which has been leveraged in engineering for anomaly detection [30], clustering, and segmentation [3].

Historically, non-parametric methods have estimated LID by modeling nearby samples with statistical processes [15], gleaning nearest neighbor information [6], measuring fractal dimension [9], and calculating simplex skewness [12]. While these methods are effective in simple, small-scale scenarios,

---

[†]Work done during an internship at Pacific Northwest National Laboratory (PNNL).

they require large amounts of sampled data, are strongly affected by hyperparameter choice, and fail to generalize in low-data settings [25, 24, 13, 28].

Recent works have estimated LID using parametric deep generative models – they inherit the advantages of deep learning: scalability to big problems and generalization to unseen data. Harnessing the power of deep generative models has led to unprecedented LID estimation capabilities on complex synthetic manifolds and applicability to high-dimensional, real-world problems [25, 24, 13].

Our work provides the following contributions:

- We prove that the denoising score matching loss is lower bounded by the LID, motivating its use as a scalable LID estimator that does not require exhaustive samples or gradient.

- We demonstrate a close relationship between the score matching losses and the current leading estimators, FLIPD [13] and the normal bundle method [24]. We prove that expected FLIPD is also lower bounded by the LID through its connection to the *implicit* score matching loss.

- We provide experiments on a manifold benchmark and with Stable Diffusion 3.5 and Stable Diffusion 2 which show that the denoising score matching loss is a highly competitive LID estimator. Moreover, it exhibits superior scalability in terms of memory footprint and consistency under model quantization.

2   Background and Related Work

Denoising Generative Models   Diffusion and flow-based models have achieved state-of-the-art generative modeling capabilities by learning to denoise data [10, 5]. The connection between denoising and generative modeling was proven by Vincent [26], who showed that the denoising objective is a scalable method to learn the *score* of the probability density ($\nabla \log p(\tilde{\mathbf{x}})$) of noised samples $\tilde{\mathbf{x}} \leftarrow \mathbf{x} + \sigma\epsilon$, where $\mathbf{x} \sim p(\mathbf{x})$, $\sigma \in \mathbb{R}^+$, and ϵ is drawn from a standard Gaussian distribution. For a score function $s_\theta : \mathbb{R}^n \to \mathbb{R}^n$ parameterized by θ, the denoising objective is equivalent (up to a constant) to the explicit score matching and implicit score matching objectives [11] for the noised distribution:

$$\mathbb{E}_{\tilde{\mathbf{x}}}\left[\mathcal{L}_{\text{ESM}}(\tilde{\mathbf{x}}, \sigma, \theta)\right] = \mathbb{E}_{\tilde{\mathbf{x}}}\left[\mathcal{L}_{\text{ISM}}(\tilde{\mathbf{x}}, \sigma, \theta)\right] + C_{\text{ISM}} = \mathbb{E}_{\mathbf{x}}\left[\mathcal{L}_{\text{DSM}}(\mathbf{x}, \sigma, \theta)\right] + C_{\text{DSM}}, \quad (1)$$

where each loss $\mathcal{L}(\boldsymbol{x}, \sigma, \theta)$ is the pointwise version of the loss evaluated at the point $\boldsymbol{x}$, and C_{ISM} and C_{DSM} are constants which do not depend on the parameters θ. The denoising objective $\mathcal{L}_{\text{DSM}}$ is particularly appealing for training deep generative models, as it does not require oracular knowledge of the true score (required in $\mathcal{L}_{\text{ESM}}$) or expensive computation of $\nabla_{\boldsymbol{x}} \cdot s_\theta(\boldsymbol{x})$ (required in $\mathcal{L}_{\text{ISM}}$). Once the score is approximated for many distributions bridging the data distribution and an easy to sample distribution (typically Gaussian), one may employ reverse diffusion samplers [23], ODE solvers [16], or Langevin dynamics [22] to draw samples from the data distribution.

Non-Parametric LID Estimation   Non-parametric local intrinsic dimension estimation methods provide data-driven approaches to estimate the structural dimensionality of high-dimensional datasets without assuming specific parametric forms. The MLE method by Levina and Bickel [15] models the distribution of distances from each point to its k nearest neighbors, estimating intrinsic dimension through maximum likelihood fitting of Poisson distribution parameters to ratios of consecutive nearest neighbor distances. Similarly exploiting nearest neighbor statistics, the TwoNN (Two Nearest Neighbors) [6] method analyzes the ratio of distances to the second and first nearest neighbors, using the empirical distribution of these ratios to infer local dimensionality. Taking a more geometric approach, Expected Simplex Skewness (ESS) [12] estimates dimension by measuring the skewness of volumes of simplices formed by points and their nearest neighbors. Beyond these smooth manifold approaches, fractal dimension methods such as box-counting and correlation dimension estimators [9] handle datasets with self-similar or fractal structure.

Parametric LID Estimation   Parametric estimators of local intrinsic dimension typically leverage deep generative models such as diffusion models, flow-matching models, or normalizing flows. LIDL [25], originally developed with normalizing flows, uses an ensemble of generative models trained on different levels of Gaussian noise. The generative models provide a set of density estimates at a point $\boldsymbol{x}$ from which the LID can be retrieved by measuring the rate of change of density estimates under

increasing noise perturbations. The normal bundle (NB) method [24] estimates LID through the connection between the score and the vector to a manifold. The NB method adds m scaled Gaussian noise instances to a point $x \in \mathbb{R}^n$ to yield $\tilde{\mathbf{x}}$ and computes the score estimate $s_\theta(\tilde{\mathbf{x}})$ for each, yielding a (m, n) matrix of score estimates. The count of non-negligible singular values is taken as the normal dimension, and that can be subtracted from the ambient dimension to yield the LID. The authors recommend using a large number of samples to ensure that there are at least as many samples as the normal dimension. Recently, Kamkari et al. [13] proposed FLIPD, which uses the Fokker-Planck equation from diffusion models to accurately estimate LID. While FLIPD shares LIDL's approach of measuring density change rates under increasing Gaussian noise, it achieves greater efficiency by requiring only a single diffusion model and one Fokker-Planck equation evaluation.

3 Score Matching as a LID Estimator

We consider training a score model on a single, sufficiently small noise level $\sigma \in \mathbb{R}^+$, and we draw connections between the denoising score matching loss and the LID of the data manifold. Let $\mathbf{x}$ be a random variable drawn from a d-dimensional data manifold $\mathcal{M}$ embedded within $\mathbb{R}^n$. Let $s_\theta : \mathbb{R}^n \to \mathbb{R}^n$ be the score function parameterized by θ. Furthermore, let ϵ be a standard Gaussian random variable in $\mathbb{R}^n$. Recall that the scaled denoising score matching loss [26] is:

$$\mathbb{E}_{\mathbf{x}} \left[\mathcal{L}_{\text{DSM}}(\mathbf{x}, \sigma, \theta) \right] := \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}), \epsilon \sim \mathcal{N}(\mathbf{0}; \mathbf{I})} \, \sigma^2 \left\| \frac{\epsilon}{\sigma} + s_\theta(\mathbf{x} + \sigma\epsilon) \right\|^2 = \mathbb{E}_{\mathbf{x}, \epsilon} \left\| \epsilon - \epsilon_\theta(\mathbf{x} + \sigma\epsilon) \right\|^2, \quad (2)$$

where $\epsilon_\theta(x) = -\sigma \, s_\theta(x)$, the noise prediction parameterization of the score.

Theorem 3.1 (Denoising Score Matching Loss Lower Bound). *Let* x, $\mathcal{L}_{DSM}$, $\mathcal{M}$, *and* d *take on the definitions above. Let* $\sigma \to 0^+$ *be sufficiently small such that the density* $p(\mathbf{x})$ *on* $\mathcal{M}$ *appears locally constant and the curvature of* $\mathcal{M}$ *is negligible over the region where the Gaussian perturbations* $\sigma\epsilon$ *have significant probability mass. Then,*

$$\mathbb{E}_{\mathbf{x}} \left[\mathcal{L}_{DSM}(\mathbf{x}, \sigma, \theta) \right] \geq d. \quad (3)$$

Please see figure (1a) for a conceptual depiction and the appendix for the proof.

Remark 3.2 (Stratified Manifolds and LID). Theorem (3.1) states that the denoising score matching loss is lower-bounded by the intrinsic dimension d of a manifold $\mathcal{M}$. Note that in the (common) case that the data comes from a stratified manifold comprised of different submanifolds $\mathcal{M}_i$ which may have different dimensions d_i, the denoising score matching loss can be written as $\mathbb{E}_{\mathcal{M}_i} \left[\mathbb{E}_{\mathbf{x} \sim \mathcal{M}_i} \mathcal{L}_{\text{DSM}}(\mathbf{x}, \sigma, \theta) \right]$. In this sense, the denoising score matching loss is lower bounded by $\mathbb{E}_{\mathcal{M}_i}[d_i]$ and the pointwise $\mathcal{L}_{\text{DSM}}(x, \sigma, \theta)$ estimates the LID of the stratified manifold.

Connection with Implicit Score Matching and FLIPD

If the LID is a lower bound on the denoising score matching loss, what does this imply for alternative losses such as *implicit* score matching? Here, we shall analyze the *implicit* score matching loss also captures the geometric properties of the data manifold $\mathcal{M}$. The *implicit* score matching loss [11] is:

$$\mathbb{E}_{\tilde{\mathbf{x}} \sim p(\tilde{\mathbf{x}})} \left[\mathcal{L}_{\text{ISM}}(\tilde{\mathbf{x}}, \sigma, \theta) \right] := \sigma^2 \, \mathbb{E}_{\tilde{\mathbf{x}} \sim p(\tilde{\mathbf{x}})} \left[\nabla \cdot s_\theta(\tilde{\mathbf{x}}) + \frac{1}{2} \| s_\theta(\tilde{\mathbf{x}}) \|^2 \right], \quad (4)$$

where $s_\theta(x) = -\sigma^{-1} \, \epsilon_\theta(x)$, $\sigma \in \mathbb{R}^+$, and $p(\tilde{\mathbf{x}}) := (p_{\mathbf{x}} * \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}))(\tilde{\mathbf{x}})$.

Theorem 3.3 (Implicit Score Matching Loss Lower Bound). *Let* x, $\mathcal{L}_{DSM}$, $\mathcal{M}$, $\tilde{\mathbf{x}}$, *and* d *take on the definitions above. Let* $\sigma \to 0^+$ *be sufficiently small such that the density* $p(\mathbf{x})$ *on* $\mathcal{M}$ *appears locally constant and the curvature of* $\mathcal{M}$ *is negligible over the region where the Gaussian perturbations* $\sigma\epsilon$ *have significant probability mass. Then,*

$$\mathbb{E}_{\tilde{\mathbf{x}}} \left[\mathcal{L}_{ISM}(\tilde{\mathbf{x}}, \sigma, \theta) \right] \geq -(n - d). \quad (5)$$

Hence, under these conditions, the implicit score matching loss is lower bounded by the negative normal dimension of $\mathcal{M}$. Please see the appendix for the proof. We observe that FLIPD [13], the current SOTA in parametric LID estimation, is remarkably similar to the implicit score matching loss. In fact, FLIPD at a point x is:

$$\text{FLIPD}(x, \sigma, \theta) := \mathcal{L}_{\text{ISM}}(x, \sigma, \theta) + \frac{\sigma^2}{2} \| s_\theta(x) \|^2 + n. \quad (6)$$

3
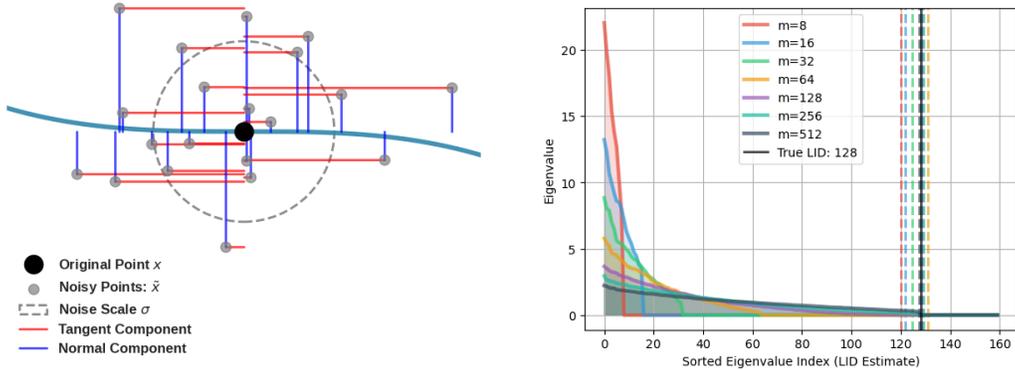
(a) Conceptual depiction of the denoising loss as a LID estimator on a uniformly sampled 1-dimensional manifold. Noise components corresponding to the tangent space yield an expected squared error of approximately 1 each, whereas noise components corresponding to the normal space yield an expected squared error of approximately 0 each. Adding up the expected squared error for each dimension yields the LID.

(b) Conceptual link between the error bundle method (solid lines) versus the denoising loss method (dashed lines). Starting from the the error Gram matrix, the error bundle method estimates the LID by counting the number of non-negligible eigenvalues, whereas the denoising loss method computes the sum of the eigenvalues (area under each spectrum). The denoising loss method is accurate at small sample sizes (e.g., $m = 8$).

Figure 1: Conceptual depiction of the denoising loss as a LID estimator (left) and a conceptual link between the denoising loss and the error bundle (EB) method.

FLIPD is calculated on noiseless data, so the additional score norm term is typically negligible. Leveraging Theorem (3.3) and the fact that $\mathbb{E}_{\tilde{\mathbf{x}}} \left[\frac{\sigma^2}{2} \|s_\theta(\tilde{\mathbf{x}})\|^2 \right] \geq 0$:

$$\mathbb{E}_{\tilde{\mathbf{x}}} \left[\text{FLIPD}(\tilde{\mathbf{x}}, \sigma, \theta) \right] \geq \mathbb{E}_{\tilde{\mathbf{x}}} \left[\mathcal{L}_{\text{ISM}}(\tilde{\mathbf{x}}, \sigma, \theta) \right] + n \geq -(n - d) + n = d. \tag{7}$$

Hence, expected FLIPD is also lower bounded by the LID through its connection to $\mathcal{L}_{\text{ISM}}$.

Connection with the Normal Bundle Estimator

Stanczuk et al. [24] propose the normal bundle (NB) LID estimator which counts the number of non-negligible singular values from a (m, n) matrix A of noise predictions $\epsilon_\theta(\tilde{\mathbf{x}})$ around a point $\boldsymbol{x}$. This is equivalent to counting the number of non-negligible eigenvalues of the (n, n) Gram matrix $C := A^T A$, in which case the eigenvalues of C are the square of the singular values of A.

Let us consider a variant of the NB estimator which operates on a (m, n) matrix B comprised of *error* vectors $(\epsilon - \epsilon_\theta(\tilde{\mathbf{x}}))$. We call this the error bundle (EB) method. Let $C' := B^T B / m$ be the Gram matrix scaled by m^{-1}. Then the number of non-negligible eigenvalues of C' should be upper bounded by d. Moreover, the sum of the eigenvalues of C' is equivalent to the denoising score matching loss at $\boldsymbol{x}$: $\text{trace}(C') = \mathcal{L}_{\text{DSM}}(\boldsymbol{x}, \sigma, \theta) \approx d$.

Figure 1b depicts this relationship for a 128 dimensional manifold in a 256 dimensional ambient space. The denoising loss is equal to the area under each of the spectra of C' calculated from increasing numbers of samples m. The denoising loss is accurate at small sample sizes (e.g., $m = 8$), whereas the EB (respectively NB) methods need at least as many samples as the LID (respectively normal dimension) to get a good estimate.

4 Experiments

We conduct a series of LID estimation experiments using manifolds of known local intrinsic dimension from the `scikit-dimension` package [1]. We use the mean absolute error (MAE) across 2000 data points of true LID versus estimated LID as our main accuracy metric. We employ MLE [15],

Table 1: LID Estimate Mean Absolute Error (MAE) on Benchmark Manifolds

Parametric	Denoising Loss (DiT)			FLIPD (DiT)		
Manifold	$\sigma = 0.01$	$\sigma = 0.02$	$\sigma = 0.05$	$\sigma = 0.01$	$\sigma = 0.02$	$\sigma = 0.05$
$d = 16\ n = 64$ HyperSphere	2.00	2.16	2.58	1.93	1.59	0.70
$d = 16\ n = 64$ HyperBall	2.57	2.59	2.71	0.27	0.51	3.69
$d = 128\ n = 256$ HyperTP	0.47	1.63	4.48	1.93	1.30	1.07
$d = 32\ n = 128$ CliffordTorus	17.47	8.47	4.18	20.15	11.16	3.30
$d = 32\ n = 128$ Nonlinear	12.47	7.54	2.15	26.54	20.44	12.87
Average	7.00	4.48	**3.22**	10.16	7.00	4.32
Parametric	Denoising Loss (MLP)			FLIPD (MLP)		
Manifold	$\sigma = 0.01$	$\sigma = 0.02$	$\sigma = 0.05$	$\sigma = 0.01$	$\sigma = 0.02$	$\sigma = 0.05$
$d = 16\ n = 64$ HyperSphere	2.06	2.07	2.09	0.67	1.20	0.95
$d = 16\ n = 64$ HyperBall	2.66	2.66	2.69	0.35	0.07	0.20
$d = 128\ n = 256$ HyperTP	0.74	0.55	0.42	1.27	7.65	3.02
$d = 32\ n = 128$ CliffordTorus	28.90	28.87	28.76	32.02	33.03	30.50
$d = 32\ n = 128$ Nonlinear	10.67	6.95	3.44	19.69	14.16	7.32
Average	9.01	8.22	7.48	10.80	11.22	8.40
Non-Parametric	MLE		TwoNN		ESS	
Manifold	$k = 50$	$k = 100$	$k = 50$	$k = 100$	$k = 50$	$k = 100$
$d = 16\ n = 64$ HyperSphere	3.18	3.94	3.99	3.53	0.49	0.32
$d = 16\ n = 64$ HyperBall	3.55	4.44	4.28	3.65	0.71	0.61
$d = 128\ n = 256$ HyperTP	79.02	84.89	83.0	78.24	7.05	2.47
$d = 32\ n = 128$ CliffordTorus	3.46	3.24	4.54	3.11	29.67	30.85
$d = 32\ n = 128$ Nonlinear	11.68	13.55	12.74	11.86	1.30	1.37
Average	20.18	22.01	21.71	20.08	7.84	7.12

TwoNN [6], and ESS [12] (from `scikit-dimension`) with $k = 50$ and $k = 100$ nearest neighbors. We train a diffusion transformer (DiT) [18] architecture with a patch size of 4, hidden dimension of 128, 16 attention heads, and 8 layers on each of the manifolds using a flow matching objective [16]. We also train an MLP with skip connections, akin to [13]. Each manifold is represented by 2000 uniformly sampled points and the model is trained for 50000 batches of size 100 using a cosine annealed learning rate schedule. We convert model output (flow predictions) to noise predictions via the parameterization presented by Esser et al. [5]. We use the same trained model to provide LID estimates using the denoising loss and using FLIPD at $\sigma = 0.01$, $\sigma = 0.02$, and $\sigma = 0.05$. Note that prior to estimating LID with either the denoising loss or FLIPD, the data must be scaled according to the schedule the flow matching model was trained on (scaled by $(1 - \sigma)$ in our experiments). We employ 8 Gaussian noise samples for the denoising loss method, and we employ 8 Rademacher samples to compute the divergence estimate for FLIPD. All experiments are implemented in PyTorch [17] and run on a single NVIDIA H100 80GB GPU. For the Clifford torus, we randomly permute the dimensions of the data such that the patch-based DiT architecture must use attention to learn the manifold's structure. We center all manifold data and scale it with σ_A^{-1}, where $\sigma_A := \max_i \sigma_i$ and σ_i is the standard deviation of manifold data feature i.

Table (1) depicts the results of the LID benchmark experiment. The non-parametric estimators perform well (MAE < 5) on low-dimensional manifolds such as the 16-HyperSphere and 16-HyperBall, however their performance drops on highly curved, high-dimensional manifolds such as the 128-dimensional HyperTwinPeaks, 32-dimensional Clifford torus, and the 32-dimensional 'Nonlinear' manifold. Of the non-parametric methods, ESS performed best with average MAE of 7.84 and 7.12 for $k = 50$ and $k = 100$, respectively.

In all combinations of architecture (MLP or DiT) and σ, the denoising loss method outperforms FLIPD in terms of average MAE. The difference in MAE between the denoising loss and FLIPD is small in most cases, but the differences can be significant on highly-curved manifolds such
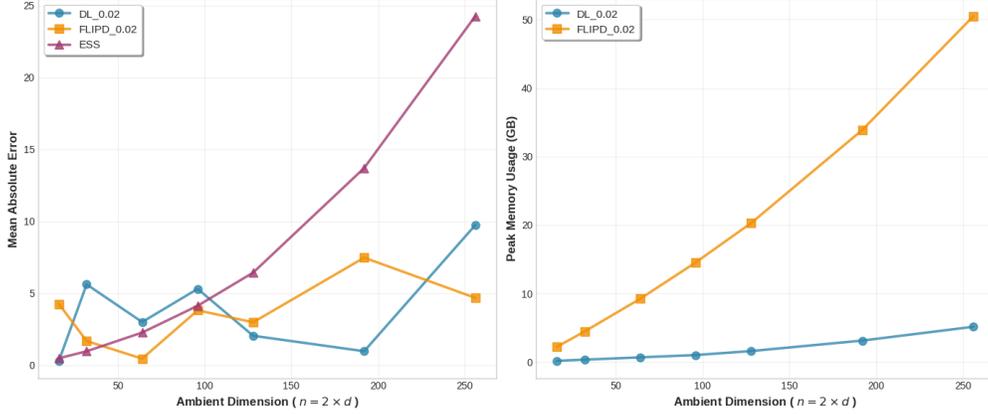
Figure 2: Comparison of LID estimation mean absolute error (left) and peak memory usage (right) of LID estimators as ambient dimension and true LID of a HyperSphere manifold increase.

as 'Nonlinear'. We hypothesize that this could be due to the impact of manifold curvature on the divergence term in FLIPD. Overall, the parametric methods leveraging the DiT architecture outperformed the non-parametric methods, with the denoising loss achieving an average MAE of **3.22** and FLIPD achieving an average MAE of 4.32 for the hyperparameter choice $\sigma = 0.05$.

In figure (2) we compare the MAE and peak GPU memory usage of FLIPD and the denoising loss for a sequence of hypersphere manifolds of increasing LID and ambient dimension. We use the DiT architecture in this experiment. For each manifold, the ambient dimension is twice the true LID of the manifold. Both FLIPD and the denoising loss (left) achieve low LID MAE as the hypersphere grows. ESS, a non-parametric method, has low MAE on small manifolds, but it fails to scale to larger manifolds (MAE ≈ 25 when $n = 256$ and $d = 128$). On the right, we compare peak GPU memory usage for simultaneous LID computation on all 2000 data samples for the two parametric methods. As the LID and ambient dimension of the space increases, the peak GPU memory usage for the parametric methods increases due to the use of higher dimensional data and larger DiT models. The memory usage of FLIPD increases rapidly due to its reliance on gradient computation. The peak memory usage of the denoising loss method grows slowly, as it does not leverage gradient computation.

Stable Diffusion 3.5 Experiments

Next, we implement the denoising loss method and FLIPD for LID estimation with the rectified flow transformer Stable Diffusion 3.5 medium (SD-3.5) [5] implemented in the `diffusers` library [27]. We sample 500 256×256 images of "a photo of a cat" from SD-3.5 with 28 sampling steps and a guidance level of 3.5. We use the null prompt "" and the noise parameterization from Esser et al. [5] for LID estimates.

Figure (3a) depicts the distributions of LID estimates for 100 of the images at varying noise scales (flow matching time). Note that the LID is estimated in latent space and that the latents are scaled by $(1 - \sigma)$ for each noise level. On average, FLIPD estimates are higher than denoising loss estimates at each noise scale. At low noise scales, the data appear relatively high dimensional and occupy most of the 16384 dimensions of the perceptually compressed latent space. Note that this is still a fraction ($< 8.4\%$) of the ambient dimension of the images. At high noise scales, the scaled data appear as a 0-dimensional point. Figure (3b) depicts a scatter plot of LID estimates with FLIP versus LID estimates with denoising loss at various noise scales. The FLIPD and denoising loss estimates are highly correlated, and the line of best fit for each noise scale has a slope < 1 due to FLIPD's higher-on-average LID estimates.

Next, we quantize SD-3.5 (float16 and bfloat16) and record change in LID estimates (measured as MAE from float32) and peak GPU memory usage (on a batch of 10 images) for the two parametric methods. We average results across all 500 images in this case. In figure (4a) we observe that the change in LID estimates for the denoising loss is lower than the change in FLIPD estimates after
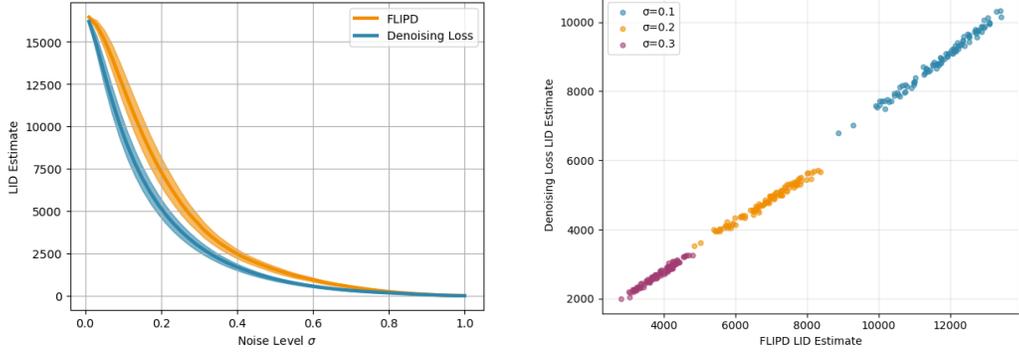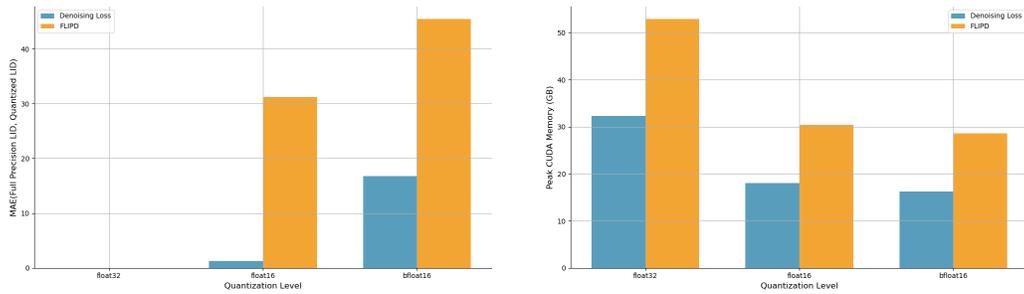
(a) $\mathcal{L}_{\text{DSM}}$ and FLIPD LID estimates on 100 256×256 images using SD-3.5.

(b) $\mathcal{L}_{\text{DSM}}$ and FLIPD LID estimate scatter plot using SD-3.5. The LID estimates are highly correlated.

Figure 3: FLIPD and $\mathcal{L}_{\text{DSM}}$ LID estimates for 100 256×256 images using SD3.5-medium. The LID estimates are highly correlated, with $\mathcal{L}_{\text{DSM}}$ providing lower estimates on average.

quantization. We hypothesize that this is due to accumulated error in the gradient computation that was introduced by quantization. Compared with bfloat16, the float16 quantization leads to lower MAE from float32, suggesting that LID estimation with SD-3.5 benefits more from higher precision than higher dynamic range. In figure (4b) we compare the peak GPU memory usage of FLIPD and the denoising loss on batches of 10 images. In each case, the peak GPU memory consumption of the denoising loss is roughly 60% of the peak memory consumption of FLIPD. Note that a significant portion of the memory consumption can be attributed to storing SD-3.5 on the GPU, and larger batch sizes could lead to larger differences in peak memory consumption.

Stable Diffusion 2 Experiments

We include a similar experiment comparing FLIPD and the denoising loss using Stable Diffusion 2 (SD2) [20]. SD2 is a latent diffusion model which uses a U-Net [21] architecture. Figure (5) depicts the results of the experiment. Like with SD3.5, the LID estimates are highly correlated, and FLIPD typically assigns higher LID estimates at low noise levels. We note that the FLIPD LID estimates become invalid (negative) at higher noise levels. This is due to the tendency of deep neural networks (such as the U-Net) to parameterize functions with high Lipschitz constants [7, 8, 29]. We hypothesize that the negative LID estimates do not occur for FLIPD with SD3.5 due to the superior noise parameterization $\epsilon_\theta(\boldsymbol{x}, \sigma) := (1 - \sigma)v_\theta(\boldsymbol{x}, \sigma) + \boldsymbol{x}$ of flow matching models [16, 5].

(a) MAE ($\downarrow$) of full precision LID estimates versus quantized model LID estimates.

(b) Average peak CUDA memory usage ($\downarrow$) for batches of 10 256×256 images.

Figure 4: Comparison of LID prediction disagreement (MAE) and peak GPU memory usage (GB) of SD-3.5-medium at various quantization levels.
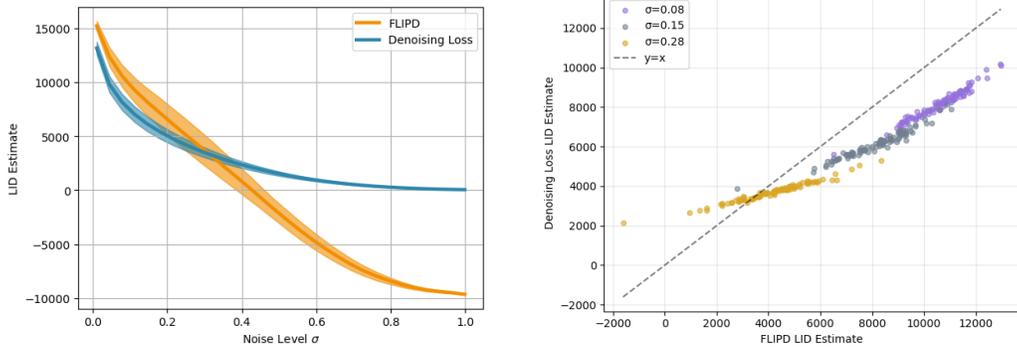
(a) $\mathcal{L}_{\text{DSM}}$ and FLIPD LID estimates on 500 512×512 images using SD2.

(b) $\mathcal{L}_{\text{DSM}}$ and FLIPD LID estimate scatter plot for 100 samples using SD2. The LID estimates are highly correlated.

Figure 5: FLIPD and $\mathcal{L}_{\text{DSM}}$ LID estimates for 512×512 images using Stable Diffusion 2 (SD2). The LID estimates are highly correlated, with $\mathcal{L}_{\text{DSM}}$ providing lower estimates at low noise levels.

5   Discussion

The Constants C_{DSM} and C_{ISM}   In this work, we show that the denoising score matching loss matches the LID of $\mathcal{M}$ in its minimum. Furthermore, recall that the minimum of $\mathcal{L}_{\text{ESM}}$ is 0 and that denoising model training involves amortizing the point-wise denoising loss over an entire dataset. Hence, the minimum of the average loss across a dataset is the average LID across the dataset, implying that the denoising constant C_{DSM} from equation (1) is the (negative) average LID of the data. Similarly, this implies that C_{ISM} is the average normal dimension of the data.

Interpretation of Multi-scale Training and Likelihood Computation   Diffusion and flow-based generative models are often trained on many data and noise scales which bridge the data distribution to a normal distribution [23, 10, 16, 5]. At each data and noise scale, one can view denoising score matching training as identifying the average LID for that particular scaled and noised manifold - see figure (3a) for visualization. Additionally, likelihood computation for both diffusion and flow-based models is based on integration of ODEs which are expressions of the negative divergence of the learned score function through the data and noise scales. Since $\mathbb{E}_{\tilde{\mathbf{x}}}\left[\mathcal{L}_{\text{ISM}}(\tilde{\mathbf{x}}, \sigma, \theta)\right] \geq -(n-d)$ from theorem (3.3), we hypothesize that one may associate higher likelihood attribution with a higher learned normal dimension (i.e., higher level of learned *structure*) at each point in the ODE solution.

Limitations   Our experiments only use 1 H100 80GB GPU and do not leverage distributed computation to support larger batch sizes. We do not quantize models beyond half precision. We do not include "knee search" on LID curves [13] and merely report average statistics for a few hyperparameters.

Societal Impacts   Our research is largely theoretical and has positive value in increasing our understanding of denoising models. We believe it does not pose any plausible negative societal impacts. The training of denoising models used in the experiments can be computationally expensive, so it caused increased electricity consumption and potential emissions.

6   Conclusion

We show that the denoising score matching loss is lower bounded by the local intrinsic dimension (LID) of the data manifold, motivating its use as a LID estimator. We show that the current leading LID estimator, FLIPD, is highly related to the implicit score matching loss, an objective which is equivalent to denoising score matching. Our experiments using a manifold benchmark indicate that the denoising loss is the most accurate LID estimator, and that it uses significantly less peak memory than FLIPD as the ambient dimension (problem size) increases. Lastly, our experiments with Stable Diffusion 3.5 show that the denoising loss requires less memory than FLIPD and exhibits less LID estimate degradation under quantization.

References

[1] Jonathan Bac, Evgeny M Mirkes, Alexander N Gorban, Ivan Tyukin, and Andrei Zinovyev. Scikit-dimension: a python package for intrinsic dimension estimation. *Entropy*, 23(10):1368, 2021.

[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[3] Kevin M Carter, Raviv Raich, and Alfred O Hero III. On local intrinsic dimension estimation and its applications. *IEEE Transactions on Signal Processing*, 58(2):650–663, 2009.

[4] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

[5] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

[6] Elena Facco, Maria d'Errico, Alex Rodriguez, and Alessandro Laio. Estimating the intrinsic dimension of datasets by a minimal neighborhood information. *Scientific reports*, 7(1):12140, 2017.

[7] Mahyar Fazlyab, Alexander Robey, Hamed Hassani, Manfred Morari, and George Pappas. Efficient and accurate estimation of lipschitz constants for deep neural networks. *Advances in neural information processing systems*, 32, 2019.

[8] Henry Gouk, Eibe Frank, Bernhard Pfahringer, and Michael J Cree. Regularisation of neural networks by enforcing lipschitz continuity. *Machine Learning*, 110(2):393–416, 2021.

[9] Peter Grassberger and Itamar Procaccia. Measuring the strangeness of strange attractors. *Physica D: nonlinear phenomena*, 9(1-2):189–208, 1983.

[10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[11] Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

[12] Kerstin Johnsson, Charlotte Soneson, and Magnus Fontes. Low bias local intrinsic dimension estimation from expected simplex skewness. *IEEE transactions on pattern analysis and machine intelligence*, 37(1):196–202, 2014.

[13] Hamid Kamkari, Brendan Ross, Rasa Hosseinzadeh, Jesse Cresswell, and Gabriel Loaiza-Ganem. A geometric view of data complexity: Efficient local intrinsic dimension estimation with diffusion models. *Advances in Neural Information Processing Systems*, 37:38307–38354, 2024.

[14] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[15] Elizaveta Levina and Peter Bickel. Maximum likelihood estimation of intrinsic dimension. *Advances in neural information processing systems*, 17, 2004.

[16] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[17] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[18] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[19] Phillip Pope, Chen Zhu, Ahmed Abdelkader, Micah Goldblum, and Tom Goldstein. The intrinsic dimension of images and its impact on learning. *arXiv preprint arXiv:2104.08894*, 2021.

[20] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[21] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[22] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[23] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[24] Jan Pawel Stanczuk, Georgios Batzolis, Teo Deveney, and Carola-Bibiane Schönlieb. Diffusion models encode the intrinsic dimension of data manifolds. In *Forty-first International Conference on Machine Learning*, 2024.

[25] Piotr Tempczyk, Rafał Michaluk, Lukasz Garncarek, Przemysław Spurek, Jacek Tabor, and Adam Golinski. Lidl: Local intrinsic dimension estimation using approximate likelihood. In *International Conference on Machine Learning*, pages 21205–21231. PMLR, 2022.

[26] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

[27] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022.

[28] Eric Yeats, Cameron Darwin, Frank Liu, and Hai Li. Adversarial estimation of topological dimension with harmonic score maps. *arXiv preprint arXiv:2312.06869*, 2023.

[29] Eric C Yeats, Yiran Chen, and Hai Li. Improving gradient regularization using complex-valued neural networks. In *International Conference on Machine Learning*, pages 11953–11963. PMLR, 2021.

[30] Fan Yin, Jayanth Srinivasa, and Kai-Wei Chang. Characterizing truthfulness in large language model generations with local intrinsic dimension. *arXiv preprint arXiv:2402.18048*, 2024.

A Proofs

A.1 Proof for Theorem 3.1: Denoising Score Matching Loss Lower Bound

Proof. We adopt the symbol definitions from the paper. Let $\mathbf{x}$ be a random variable drawn from a d-dimensional data manifold $\mathcal{M}$ embedded within $\mathbb{R}^n$. Let $s_\theta : \mathbb{R}^n \to \mathbb{R}^n$ be a score function parameterized by θ. Furthermore, let ϵ be a standard Gaussian random variable in $\mathbb{R}^n$ and $\sigma \in \mathbb{R}^+$. Leveraging the parameterization $\epsilon_\theta(\boldsymbol{x}) = -\sigma \, s_\theta(\boldsymbol{x})$, the scaled denoising score matching loss is:

$$\mathbb{E}_\mathbf{x}\left[\mathcal{L}_{\text{DSM}}(\mathbf{x}, \, \sigma, \, \theta)\right] := \mathbb{E}_{\mathbf{x},\epsilon}\left[\|\epsilon - \epsilon_\theta(\mathbf{x} + \sigma\epsilon)\|^2\right]. \tag{8}$$

For each $\boldsymbol{x} \in \mathcal{M}$, let $T_{\boldsymbol{x}}\mathcal{M}$ be the tangent space of $\mathcal{M}$ at the point $\boldsymbol{x}$ and $N_{\boldsymbol{x}}\mathcal{M}$ be the normal space of $\mathcal{M}$ at the point $\boldsymbol{x}$. Since $\mathcal{M}$ is d-dimensional, there exists an orthonormal basis $\{u_{\boldsymbol{x},1}, \ldots, u_{\boldsymbol{x},d}, v_{\boldsymbol{x},1}, \ldots, v_{\boldsymbol{x},n-d}\}$ such that $\text{span}\{u_{\boldsymbol{x},1}, \ldots, u_{\boldsymbol{x},d}\} = T_{\boldsymbol{x}}\mathcal{M}$ and $\text{span}\{v_{\boldsymbol{x},1}, \ldots, v_{\boldsymbol{x},n-d}\} = N_{\boldsymbol{x}}\mathcal{M}$ for each $\boldsymbol{x} \in \mathcal{M}$. We may re-write the denoising score matching loss using the bases $T_{\boldsymbol{x}}\mathcal{M}$ and $N_{\boldsymbol{x}}\mathcal{M}$ at each point:

$$\mathbb{E}_\mathbf{x}\left[\mathcal{L}_{\text{DSM}}(\mathbf{x}, \, \sigma, \, \theta)\right] = \mathbb{E}_{\mathbf{x},\epsilon}\left[\left(\sum_{i=1}^d \|\langle\epsilon, u_{\mathbf{x},i}\rangle - \langle\epsilon_\theta(\mathbf{x} + \sigma\epsilon), u_{\mathbf{x},i}\rangle\|^2\right)\right.$$
$$\left. + \left(\sum_{i=1}^{n-d} \|\langle\epsilon, v_{\mathbf{x},i}\rangle - \langle\epsilon_\theta(\mathbf{x} + \sigma\epsilon), v_{\mathbf{x},i}\rangle\|^2\right)\right]. \tag{9}$$

For the remainder of the proof, we use the shorthand notation $\epsilon_i := \langle\epsilon, u_{\mathbf{x},i}\rangle$ or $\epsilon_i := \langle\epsilon, v_{\mathbf{x},i}\rangle$, depending on the context. Similarly, we use $\epsilon_\theta(\mathbf{x} + \sigma\epsilon)_i := \langle\epsilon_\theta(\mathbf{x} + \sigma\epsilon), u_{\mathbf{x},i}\rangle$ or $\epsilon_\theta(\mathbf{x} + \sigma\epsilon)_i := \langle\epsilon_\theta(\mathbf{x} + \sigma\epsilon), v_{\mathbf{x},i}\rangle$, depending on the context. We omit the dependence of u or v on $\mathbf{x}$ for simplicity. Moreover, we define $\tilde{\mathbf{x}} = \mathbf{x} + \epsilon$ as the 'noised' version of $\mathbf{x}$.

Lemma A.1 (Mean Squared Error Bound). *The mean squared error of the Gaussian variable ϵ_i with its estimator $\epsilon_\theta(\tilde{\mathbf{x}})_i$ is lower bounded by the entropy power [4]:*

$$\mathbb{E}_{\epsilon,\tilde{\mathbf{x}}}\|\epsilon_i - \epsilon_\theta(\tilde{\mathbf{x}})_i\|^2 \geq \frac{1}{2\pi e}e^{2h(\epsilon_i|\tilde{\mathbf{x}})}. \tag{10}$$

This bound on estimator error forms the basis for the bound on the denoising score matching loss. The conditional differential entropy $h(\epsilon_i|\tilde{\mathbf{x}})$ is:

$$h(\epsilon_i|\tilde{\mathbf{x}}) = \int_{\tilde{\boldsymbol{x}}} p(\tilde{\boldsymbol{x}}) \int_{\epsilon_i} -p(\epsilon_i|\tilde{\boldsymbol{x}}) \log p(\epsilon_i|\tilde{\boldsymbol{x}}) \, \mathrm{d}\epsilon_i \, \mathrm{d}\tilde{\boldsymbol{x}}. \tag{11}$$

Leveraging the identity $p(\tilde{\mathbf{x}}) = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} p(\tilde{\mathbf{x}}|\mathbf{x})$ and distributing, we have:

$$h(\epsilon_i|\tilde{\mathbf{x}}) = \int_{\boldsymbol{x}} p(\boldsymbol{x}) \int_{\tilde{\boldsymbol{x}}} p(\tilde{\boldsymbol{x}}|\boldsymbol{x}) \int_{\epsilon_i} -p(\epsilon_i|\tilde{\boldsymbol{x}}) \log p(\epsilon_i|\tilde{\boldsymbol{x}}) \, \mathrm{d}\epsilon_i \, \mathrm{d}\tilde{x} \, \mathrm{d}\boldsymbol{x}. \tag{12}$$

Here, $\boldsymbol{x} \sim p(\boldsymbol{x})$ yields $T_{\boldsymbol{x}}\mathcal{M}$ and $N_{\boldsymbol{x}}\mathcal{M}$, and $p(\tilde{\boldsymbol{x}}|\boldsymbol{x})$ generates a (sufficiently small) neighborhood around each $\boldsymbol{x}$ such that $T_{\boldsymbol{x}}\mathcal{M}$ and $N_{\boldsymbol{x}}\mathcal{M}$ are valid for $\tilde{\mathbf{x}}$ and $p(\boldsymbol{x})$ is locally constant on $\mathcal{M}$ with probability one. The innermost term $\int_{\epsilon_i} -p(\epsilon_i|\tilde{\boldsymbol{x}}) \log p(\epsilon_i|\tilde{\boldsymbol{x}}) \, \mathrm{d}\epsilon_i$ is sensitive to whether ϵ_i lies in the tangent or normal space at $\boldsymbol{x}$, connecting the bound to the LID. We apply Bayes' theorem:

$$p(\epsilon_i|\tilde{\mathbf{x}}) = \frac{p(\epsilon_i) \, p(\tilde{\mathbf{x}}|\epsilon_i)}{p(\tilde{\mathbf{x}})}. \tag{13}$$

Leveraging the fact that $\tilde{\mathbf{x}} = \mathbf{x} + \sigma\epsilon$ and the identity $p(\tilde{\boldsymbol{x}}|\epsilon_i) = \int_{\boldsymbol{x}} p(\tilde{\boldsymbol{x}}|\boldsymbol{x}, \epsilon_i)p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$, we discern that $p(\tilde{\mathbf{x}}|\epsilon_i)$ is the result of a convolution of $p(\mathbf{x})$ with the product of a scaled Gaussian distribution and a Dirac delta in the i-th (tangent or normal) dimension. Hence,

$$p(\tilde{\mathbf{x}}|\epsilon_i) = \int_{\boldsymbol{y}} p_\mathbf{x}(\mathbf{x} - \boldsymbol{y}) \left(\delta(\boldsymbol{y}_i - \sigma\epsilon_i) \prod_{j \neq i} \mathcal{N}(\boldsymbol{y}_j; 0, \sigma^2)\right) d\boldsymbol{y} \tag{14}$$

With $\boldsymbol{x}$ fixed and with $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \sigma\epsilon_i$, the distributions $p(\epsilon_i)$ and $p(\tilde{\boldsymbol{x}})$ from equation (13) are asymptotically similar with varying ϵ_i. In other words, the ratio $\frac{p(\epsilon_i)}{p(\tilde{\boldsymbol{x}})}$ from (13) is finite in the tails. Leveraging this, we first consider the case that $\epsilon_i \in N_{\boldsymbol{x}}\mathcal{M}$.

11

Case 1: $\epsilon_i \in N_{\boldsymbol{x}}\mathcal{M}$. If ϵ_i lies in the **normal space**, $p(\tilde{\mathbf{x}}|\epsilon_i) \approx p(\tilde{\mathbf{x}})$ in every dimension *except* v_i. $p(\tilde{\mathbf{x}}|\epsilon_i)$ is a *translation* of $p(\mathbf{x})$ in v_i due to the Dirac delta $\delta(\boldsymbol{y}_i - \sigma\epsilon_i)$ from (14). Because the distribution $p(\mathbf{x}) \in \mathcal{M}$ appears as a Dirac delta in $N_{\boldsymbol{x}}\mathcal{M}$, the differential entropy along v_i $\int_{\epsilon_i} -p(\epsilon_i|\tilde{\boldsymbol{x}}) \log p(\epsilon_i|\tilde{\boldsymbol{x}}) \, \mathrm{d}\epsilon_i = -\infty$. Plugging this into lemma (A.1), we yield

$$\mathbb{E}_{\epsilon, \tilde{\mathbf{x}}} \|\epsilon_i - \epsilon_\theta(\tilde{\mathbf{x}})_i\|^2 \geq \frac{1}{2\pi e} e^{2\mathbb{E}_{\mathbf{x}, \tilde{\mathbf{x}}} - \infty} = 0. \tag{15}$$

Case 2: $\epsilon_i \in T_{\boldsymbol{x}}\mathcal{M}$. If ϵ_i lies in the **tangent space**, $p(\tilde{\mathbf{x}}|\epsilon_i) \approx p(\tilde{\mathbf{x}})$ in the neighborhood, and the ratio $\frac{p(\tilde{\mathbf{x}}|\epsilon_i)}{p(\tilde{\mathbf{x}})}$ is finite in the tails. $p(\epsilon_i|\tilde{\mathbf{x}})$ is therefore dominated by $p(\epsilon_i)$, a 1D standard Gaussian distribution with differential entropy $\frac{1}{2} \log 2\pi e$. Plugging this into lemma (A.1), we yield

$$\mathbb{E}_{\epsilon, \tilde{\mathbf{x}}} \|\epsilon_i - \epsilon_\theta(\tilde{\mathbf{x}})_i\|^2 \geq \frac{1}{2\pi e} e^{2\mathbb{E}_{\mathbf{x}, \tilde{\mathbf{x}}} \frac{1}{2} \log 2\pi e} = 1. \tag{16}$$

Combining **Case 1** and **Case 2** with (9), we yield the statement of the theorem:

$$\mathbb{E}_{\mathbf{x}} \left[\mathcal{L}_{\text{DSM}}(\mathbf{x}, \sigma, \theta) \right] \geq d. \tag{17}$$

$\square$

A.2 Proof for Theorem 3.3: Implicit Score Matching Loss Lower Bound

Proof. We adopt the symbol definitions from the paper. Let $\mathbf{x}$ be a random variable drawn from a uniformly distributed d-dimensional data manifold $\mathcal{M}$ embedded within $\mathbb{R}^n$. Let $s_\theta : \mathbb{R}^n \to \mathbb{R}^n$ be a score function parameterized by θ. Furthermore, let ϵ be a standard Gaussian random variable in $\mathbb{R}^n$ and $\sigma \in \mathbb{R}^+$. With the random variable $\tilde{\mathbf{x}} = \mathbf{x} + \sigma\epsilon$, the scaled implicit score matching loss is:

$$\mathbb{E}_{\tilde{\mathbf{x}} \sim p(\tilde{\mathbf{x}})} \left[\mathcal{L}_{\text{ISM}}(\tilde{\mathbf{x}}, \sigma, \theta) \right] := \sigma^2 \, \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \mathbb{E}_{\tilde{\mathbf{x}} \sim p(\tilde{\mathbf{x}}|\mathbf{x})} \left[\nabla \cdot s_\theta(\tilde{\mathbf{x}}) + \frac{1}{2} \|s_\theta(\tilde{\mathbf{x}})\|^2 \right]. \tag{18}$$

Recall from (1) that the minimizer θ^* of $\mathbb{E}_{\tilde{\mathbf{x}}} \left[\mathcal{L}_{\text{ESM}}(\tilde{\mathbf{x}}, \sigma, \theta) \right]$ also minimizes $\mathbb{E}_{\tilde{\mathbf{x}}} \left[\mathcal{L}_{\text{ISM}}(\tilde{\mathbf{x}}, \sigma, \theta) \right]$ and satisfies $s_{\theta^*}(x) = \nabla \log p(x)$ almost everywhere [11].

For each $\boldsymbol{x} \in \mathcal{M}$, let $T_{\boldsymbol{x}}\mathcal{M}$ be the tangent space of $\mathcal{M}$ at the point $\boldsymbol{x}$ and $N_{\boldsymbol{x}}\mathcal{M}$ be the normal space of $\mathcal{M}$ at the point $\boldsymbol{x}$. Since $\mathcal{M}$ is d-dimensional, there exists an orthonormal basis $\{u_{\boldsymbol{x},1}, \ldots, u_{\boldsymbol{x},d}, v_{\boldsymbol{x},1}, \ldots, v_{\boldsymbol{x},n-d}\}$ such that $\text{span}\{u_{\boldsymbol{x},1}, \ldots, u_{\boldsymbol{x},d}\} = T_{\boldsymbol{x}}\mathcal{M}$ and $\text{span}\{v_{\boldsymbol{x},1}, \ldots, v_{\boldsymbol{x},n-d}\} = N_{\boldsymbol{x}}\mathcal{M}$ for each $\boldsymbol{x} \in \mathcal{M}$. The remainder of the proof will show that as $\sigma \to 0^+$ the minimum implicit score matching loss is equivalent to the negative normal dimension of the manifold $\mathcal{M}$, yielding the lower bound.

We assume $\sigma \to 0^+$ is sufficiently small such that in the neighborhood defined by $p(\tilde{\mathbf{x}}|\mathbf{x})$, $p(\mathbf{x})$ is locally constant on $\mathcal{M}$ and the curvature of $\mathcal{M}$ is negligible. Therefore, the neighborhood samples $\tilde{\mathbf{x}} \sim p(\tilde{\mathbf{x}}|\mathbf{x})$ inherit $T_{\mathbf{x}}\mathcal{M}$ and $N_{\mathbf{x}}\mathcal{M}$ from the observation of $\mathbf{x}$. Leveraging this, let us re-write the implicit score matching loss (18) with optimal parameters θ^* along the components $u_{\mathbf{x},\cdot}$ of $T_{\mathbf{x}}\mathcal{M}$ and $v_{\mathbf{x},\cdot}$ of $N_{\mathbf{x}}\mathcal{M}$:

$$\mathbb{E}_{\tilde{\mathbf{x}} \sim p(\tilde{\mathbf{x}})} \left[\mathcal{L}_{\text{ISM}}(\tilde{\mathbf{x}}, \sigma, \theta^*) \right] =$$

$$\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \mathbb{E}_{\tilde{\mathbf{x}} \sim p(\tilde{\mathbf{x}}|\mathbf{x})} \left[\sigma^2 \left(\sum_{i=1}^{d} \langle u_{\mathbf{x},i}, \nabla s_{\theta^*}(\tilde{\mathbf{x}}) u_{\mathbf{x},i} \rangle + \frac{1}{2} \langle s_\theta(\tilde{\mathbf{x}}), u_{\mathbf{x},i} \rangle^2 \right) \right.$$

$$\left. + \sigma^2 \left(\sum_{i=1}^{n-d} \langle v_{\mathbf{x},i}, \nabla s_{\theta^*}(\tilde{\mathbf{x}}) v_{\mathbf{x},i} \rangle + \frac{1}{2} \langle s_\theta(\tilde{\mathbf{x}}), v_{\mathbf{x},i} \rangle^2 \right) \right]. \tag{19}$$

Along tangent dimensions $u_{\mathbf{x},i}$, the true distribution $p(\tilde{\mathbf{x}}) = (p_{\mathbf{x}} * \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}))(\tilde{\mathbf{x}})$ is constant due to the negligible curvature (within the σ-neighborhood) and locally uniform distribution of $p_{\mathbf{x}}(\mathbf{x})$ on $\mathcal{M}$. Hence, $\langle u_{\mathbf{x},i}, \nabla s_{\theta^*}(\tilde{\mathbf{x}}) u_{\mathbf{x},i} \rangle = 0$ for each $u_{\mathbf{x},i}$. We note $\langle s_\theta(\tilde{\mathbf{x}}), u_{\mathbf{x},i} \rangle^2 \geq 0$.

Along normal dimensions $v_{\mathbf{x},i}$, the true distribution $p(\tilde{\mathbf{x}}) = (p_{\mathbf{x}} * \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}))(\tilde{\mathbf{x}})$ is proportional to $\mathcal{N}(\langle \mathbf{x}, v_{\mathbf{x},i} \rangle, \sigma^2)$. Hence, $\langle v_{\mathbf{x},i}, \nabla s_{\theta^*}(\tilde{\mathbf{x}}) v_{\mathbf{x},i} \rangle = -\sigma^{-2}$ for each $v_{\mathbf{x},i}$. We note $\langle s_\theta(\tilde{\mathbf{x}}), v_{\mathbf{x},i} \rangle^2 \geq 0$.

We plug these observations into (19), yielding the statement of the theorem:

$$\mathbb{E}_{\tilde{\mathbf{x}} \sim p(\tilde{\mathbf{x}})} \left[\mathcal{L}_{\text{ISM}}(\tilde{\mathbf{x}}, \sigma, \theta) \right] \geq \mathbb{E}_{\tilde{\mathbf{x}} \sim p(\tilde{\mathbf{x}})} \left[\mathcal{L}_{\text{ISM}}(\tilde{\mathbf{x}}, \sigma, \theta^*) \right]$$

$$\geq \mathbb{E}_{\tilde{\mathbf{x}} \sim p(\tilde{\mathbf{x}})} \left[\sigma^2 \left(\sum_{i=1}^{d}(0) + \frac{1}{2}(0) \right) + \sigma^2 \left(\sum_{i=1}^{n-d} \frac{-1}{\sigma^2} + \frac{1}{2}(0) \right) \right] = -(n - d). \quad (20)$$

$\square$

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We provide proofs and experiments to justify the claims made in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We include a limitations section in the discussion.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Our theoretical claim is supported by a proof in the main document. We clearly state if a conjecture or discussion of any theory is not supported by a formal proof.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide adequate reproducibility information including hyperparameters, methods, and choice of architecture. The new methods are simple to implement and are commonly used (for training diffusion models).

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The data used in this paper are easily retrieved from open-source repositories. We do not release the code for the experiments, however the denoising loss method is simple to implement and is already in widespread use for diffusion model training.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We provide sufficient experimental information such as hyperparameters, training details, architecture choices, and methods such that one may reproduce the results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The paper provides distributional information such as standard deviations, average statistics, and results from multiple reasonable hyperparameter choices to support our claims.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We provide information on the computing resources in the experiments section.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: Our contributions and experiments are largely theoretical and do not violate the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: While we do not foresee societal impacts which are beyond the ordinary, we addressed the potential societal impacts in the discussion section.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We cite all models, code, data, and ideas used in the paper. We use all licensed resources in accordance with the license agreement.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard component.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.