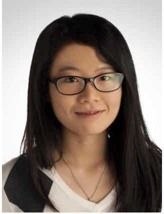
Sample Complexity of Diffusion Models for Learning Distributions on Low Dimensional Manifolds

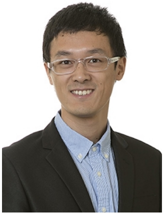
Zixuan Zhang

Georgia Tech ISyE

Oct. 2024

Joint Work with



Kaixuan Huang Mengdi Wang
Princeton Univ.

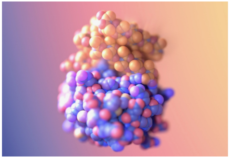
Tuo Zhao
Georgia Tech

Minshuo Chen
Northwestern Univ.

Transformative Power of Diffusion Models

DALL-E 3 by OpenAI
Image Generation

Sora by OpenAI
Video Generation



RFdiffusion by UW
Protein Design

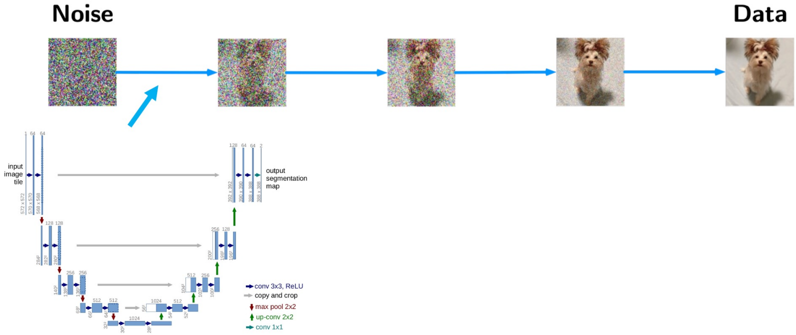
Diffusion Model in Generation

- Generate samples from noise.
- Sequential transformation.

(Sohl-Dickstein et. al., 2015, Song and Ermon, 2019, Ho et. al., 2020)

Diffusion Model in Generation

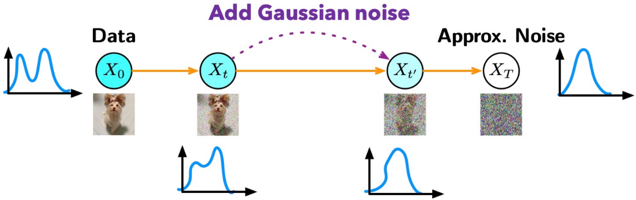
- Generate samples from noise.
- Sequential transformation.



(Sohl-Dickstein et. al., 2015, Song and Ermon, 2019, Ho et. al., 2020)

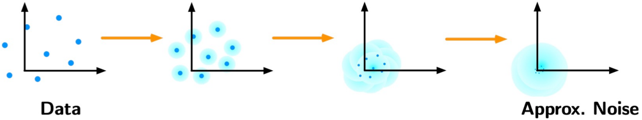
Forward Process - Noise Corruption

- Noise corruption process



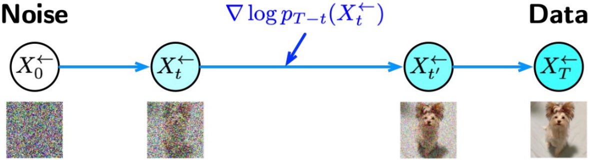
$$\mathrm{d}X_t = -\frac{1}{2}X_t\mathrm{d}t + \mathrm{d}W_t$$

- Data distribution transformed into centered Gaussian



Backward Process - Sample Generation

- Time reversal in distribution



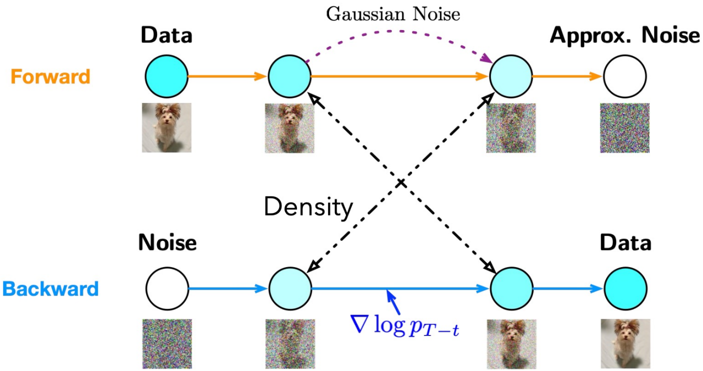
Forward Process $\quad \mathrm{d}X_t = -\frac{1}{2}X_t \mathrm{d}t + \mathrm{d}W_t$

Backward Process $\quad \mathrm{d}X_t^{\leftarrow} = \left[\frac{1}{2}X_t + \nabla \log p_{T-t}(X_t^{\leftarrow})\right]\mathrm{d}t + \mathrm{d}\bar{W}_t$

Score function

(Anderson, 1982; Haussmann and Pardoux, 1986)

Forward and Backward Coupling



Success Despite Curse of Dimensionality

- Sample size (Niles-Weed and Berthet, 2022).

$$\#\text{samples} \asymp \epsilon^{-\frac{D+2s}{s+1}}.$$

- ImageNet resolution: $D = 224 \times 224 \times 3$.

$$\#\text{samples} \geq 10^{224\times 224}.$$

- However, diffusion models are trained with $<$ 7B samples (Schuhmann et. al., 2022).

ϵ – error level; D – data dimensional; s – smoothness.

Success Despite Curse of Dimensionality

- Sample size (Niles-Weed and Berthet, 2022).

$$\#\text{samples} \asymp \epsilon^{-\frac{D+2s}{s+1}}.$$

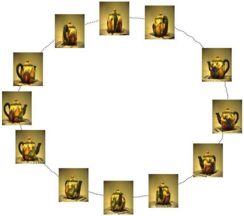
- ImageNet resolution: $D = 224 \times 224 \times 3$.

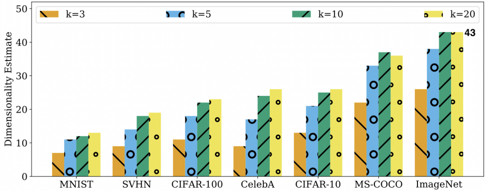
$$\#\text{samples} \geq 10^{224\times 224}.$$

- However, diffusion models are trained with $<$ 7B samples (Schuhmann et. al., 2022).

ϵ – error level; D – data dimensional; s – smoothness.

Good News: Low-Dimensional Data Structures





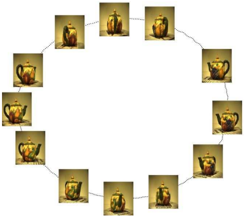
– Credit: Phillip Pope et al. ICLR 2021.

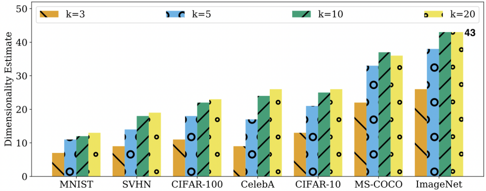
Intrinsic dimension $d \ll$ Ambient dimension D.

- Deep neural networks are adaptive in supervised learning (Chen et. al., 2022; Liu et. al., 2023; Ji et. al., 2023).
- Sample complexity scales with d instead of D.

Can we establish the **sample complexity** of diffusion models, free of curse of ambient dimensionality?

Good News: Low-Dimensional Data Structures





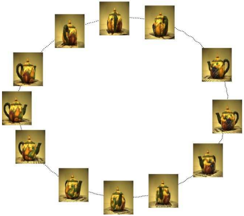
– Credit: Phillip Pope et al. ICLR 2021.

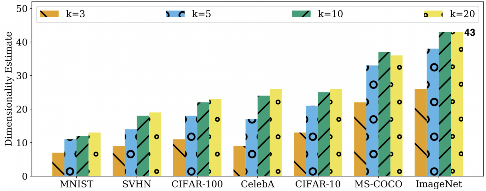
Intrinsic dimension $d \ll$ Ambient dimension D.

- Deep neural networks are adaptive in supervised learning (Chen et. al., 2022; Liu et. al., 2023; Ji et. al., 2023).
- Sample complexity scales with d instead of D.

Can we establish the **sample complexity** of diffusion models, free of curse of ambient dimensionality?

Good News: Low-Dimensional Data Structures





– Credit: Phillip Pope et al. ICLR 2021.

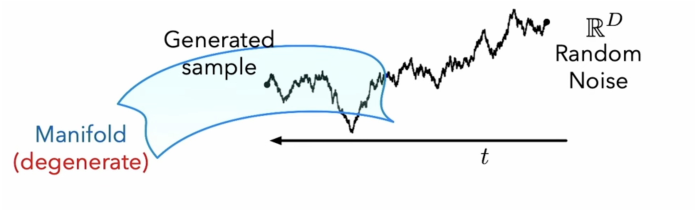
Intrinsic dimension $d \ll$ Ambient dimension D.

- Deep neural networks are adaptive in supervised learning (Chen et. al., 2022; Liu et. al., 2023; Ji et. al., 2023).
- Sample complexity scales with d instead of D.

> Can we establish the **sample complexity** of diffusion models, free of curse of ambient dimensionality?

However...

- Prior arts are not enough to explain diffusion models.
  - Diffusion model is unsupervised learning.
  - Diffusion model is a dynamic system, implemented in $\mathbb{R}^D$.

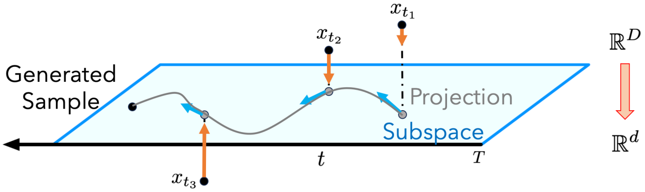


Simple but Insightful: Linear Subspace

- The score function consists of two components, on-subspace score and orthogonal score (Chen et. al., 2023).

$$\nabla \log p_t(x) = A \nabla \log p_t^z(A^\top x) - \frac{1}{1 - e^{-t}}(I_D - AA^\top)x$$

On-subspace Orthogonal

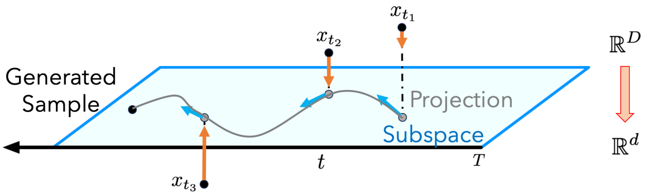


Simple but Insightful: Linear Subspace

- The score function consists of two components, on-subspace score and orthogonal score (Chen et. al., 2023).

$$\nabla \log p_t(x) = A \nabla \log p_t^z(A^\top x) - \frac{1}{1 - e^{-t}}(I_D - AA^\top)x$$

On-subspace　　　　Orthogonal

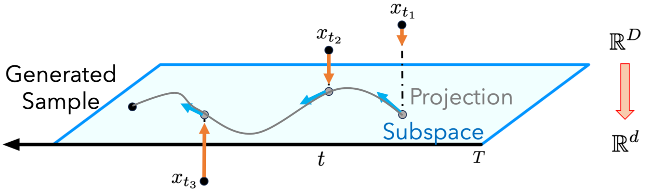


Simple but Insightful: Linear Subspace

- The score function consists of two components, on-subspace score and orthogonal score (Chen et. al., 2023).

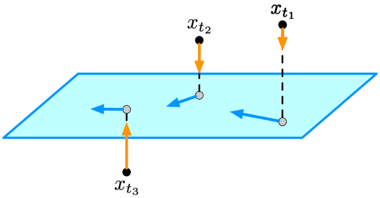
$$\nabla \log p_t(x) = A \nabla \log p_t^z(A^\top x) - \frac{1}{1 - e^{-t}} (I_D - AA^\top) x$$

On-subspace Orthogonal



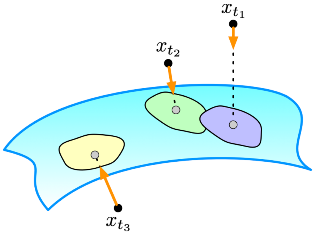
Delve into Manifolds

Linear Subspace



Score = On-subspace + Orthogonal

Manifold

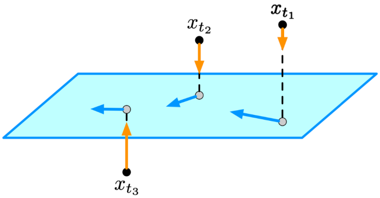


Score = On-manifold + Orthogonal
+ **interaction-term**

Curvature dependent!

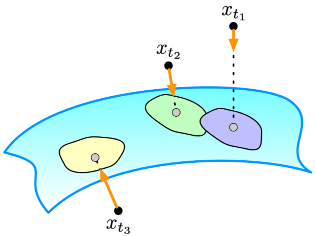
Delve into Manifolds

Linear Subspace



Score = On-subspace + Orthogonal

Manifold

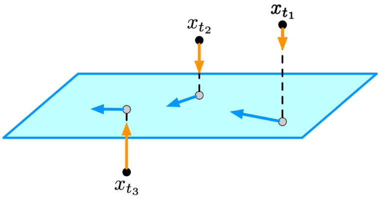


Score = On-manifold + Orthogonal
+ **interaction-term**

Curvature dependent!

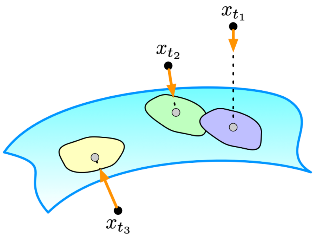
Delve into Manifolds

Linear Subspace



Score = On-subspace + Orthogonal

Manifold



Score = On-manifold + Orthogonal
+ **interaction-term**

Curvature dependent!

Score Decomposition

- Score decomposition via projection Π_t onto manifold:

$$\nabla \log p_t(x) = s_{\mathcal{M}}\left(\Pi_t(x); t\right) - \frac{1}{1-e^{-t}}\left(x - \Pi_t(x)\right) + \text{interaction}.$$

On-Manifold Orthogonal

- Orthogonal score **blows up** when time approaches zero.
- Only holds for inputs near manifold.

Score Decomposition

- Score decomposition via projection Π_t onto manifold:

$$\nabla \log p_t(x) = s_{\mathcal{M}}\left(\Pi_t(x); t\right) - \frac{1}{1-e^{-t}}\left(x - \Pi_t(x)\right) + \text{interaction}.$$

On-Manifold Orthogonal

- Orthogonal score **blows up** when time approaches zero.
- Only holds for inputs near manifold.

Score Decomposition

- Score decomposition via projection Π_t onto manifold:

$$\nabla \log p_t(x) = s_{\mathcal{M}}\left(\Pi_t(x); t\right) - \frac{1}{1-e^{-t}}\left(x - \Pi_t(x)\right) + \text{interaction}.$$

On-Manifold Orthogonal

- Orthogonal score **blows up** when time approaches zero.
- Only holds for inputs near manifold.

Score Decomposition

- Score decomposition via projection Π_t onto manifold:

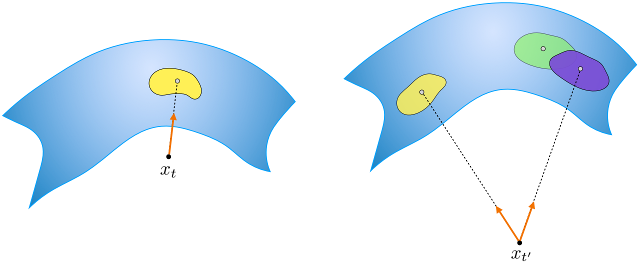
$$\nabla \log p_t(x) = s_{\mathcal{M}}\left(\Pi_t(x); t\right) - \frac{1}{1-e^{-t}}\left(x - \Pi_t(x)\right) + \text{interaction}.$$

On-Manifold Orthogonal

- Orthogonal score **blows up** when time approaches zero.
- Only holds for inputs near manifold.

Score Behavior for “Faraway” Inputs

- Score locates corrupted data to nearby neighborhoods.
- For each neighborhood, score consists of on tangent-space score and orthogonal score.



x_t close to manifold $\quad$ $x_{t'}$ far from manifold

Distribution Recovery

Theorem

Assume P_0 is supported on a d-dimensional manifold with $d \ll D$.

1. Score network (overparameterized) converges at the rate

$$\widetilde{\mathcal{O}}\left([\mathrm{curv}(\mathcal{M}) + 1]\, n^{-\frac{s}{d+2s}}\right).$$

2. Estimated distribution converges at the rate

$$W_1(\widehat{P}, P_0) = \widetilde{\mathcal{O}}\left([\mathrm{curv}(\mathcal{M}) + 1]\, n^{-\frac{s+1}{d+2s}}\right).$$

Here s is the smoothness of P_0.

- **Adaptive** to data intrinsic structures.
- **Efficient** in learning data distributions.
  Matches the minimax rate (Tang and Yang, 2022).

Summary

- Score behavior.
- NN score estimation.
- Sample complexity of distribution estimation.

Reference

[1] Chen, M., Huang, K., Zhao, T., and Wang, M. “Score approximation, estimation and distribution recovery of diffusion models on low-dimensional data”, *In International Conference on Machine Learning*, 2023.

[2] M. Chen, H. Jiang, W. Liao and Tuo Zhao, “Nonparametric Regression on Low-Dimensional Manifolds using Deep ReLU Networks”, *IMA Information and Inference*, 2021.

[3] Zhang, K., Zhang, Z., Chen, M., Takeda, Y., Wang, M., Zhao, T., and Wang, Y. X. “Nonparametric Classification on Low Dimensional Manifolds using Overparameterized Convolutional Residual Networks”, *arXiv preprint*, 2023.

[4] Liu, H., Chen, M., Zhao, T. and Liao, W. “Besov function approximation and binary classification on low-dimensional manifolds using convolutional residual networks”, *In International Conference on Machine Learning*, 2021.

Thank You!