DeepMind

Denoising Diffusion models - Generative Modeling and Inference

Arnaud Doucet

Special thanks to Valentin De Bortoli

MCM 2023 - June 28th

Generative Modeling

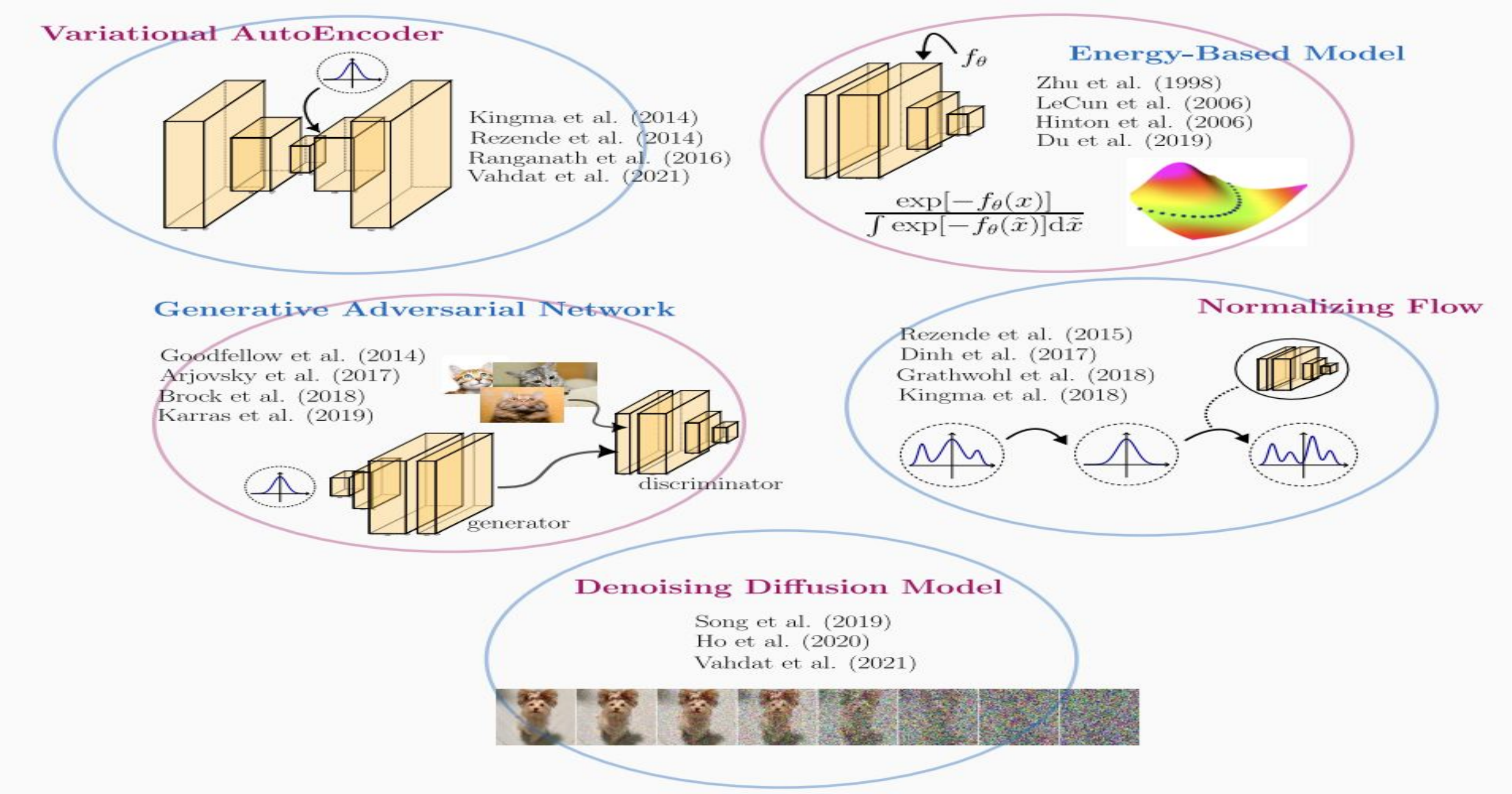
- **Formulation:** Given samples $(x_i)_{i=1}^N$ from distribution π, generate new samples distributed approximately from π.
- **Very active area** since 2010's: VAEs (Kingma & Welling, 2014), GANs (Goodfellow et al., 2014), Autogressive models (van den Oord et al., 2016).
- **Numerous applications**, e.g. data augmentation for downstream tasks (e.g. videos for self-driving cars), high-resolution nowcasting, data-driven priors for inverse problems/Bayesian inference.

From Ho et al., NeurIPS 2020

from Ravuri et al. (2021)

Generative Models

Public



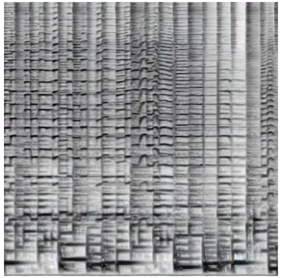
Denoising Diffusion Models

Public

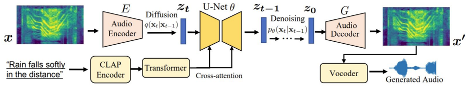
DALLE-2
OpenAI

MidJourney

Imagen
Google



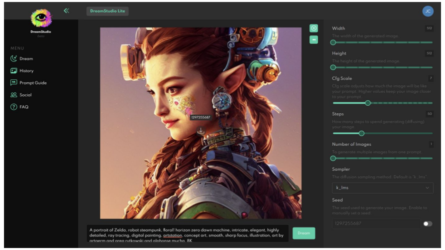
Riffusion



Make-an-audio

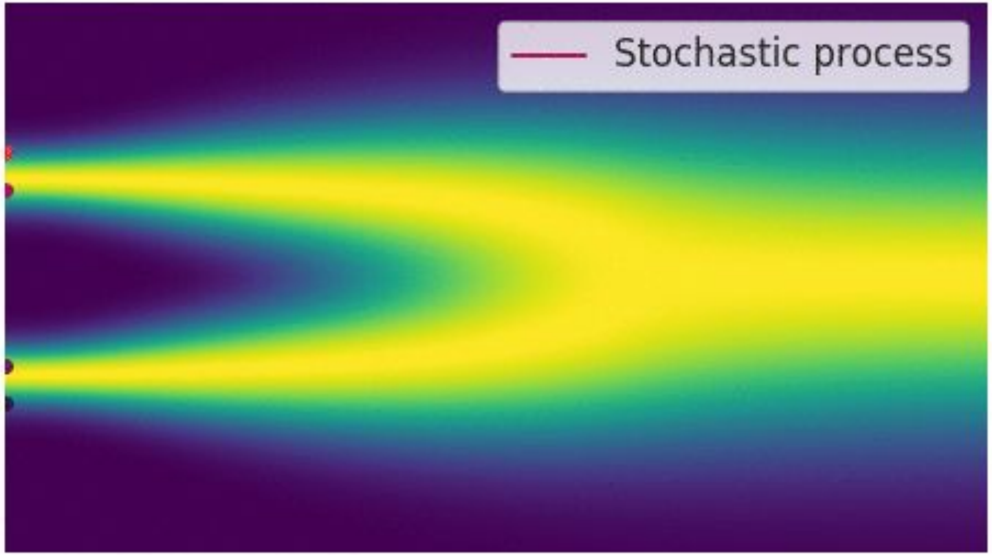
runway

Runway



Stable Diffusion

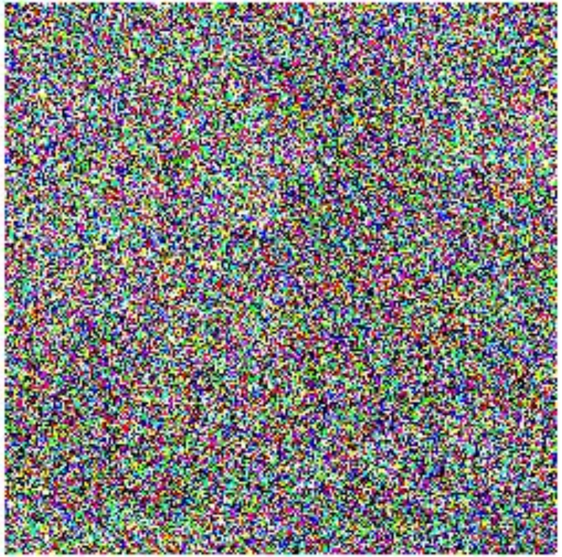
Introduction: Noising Mechanism

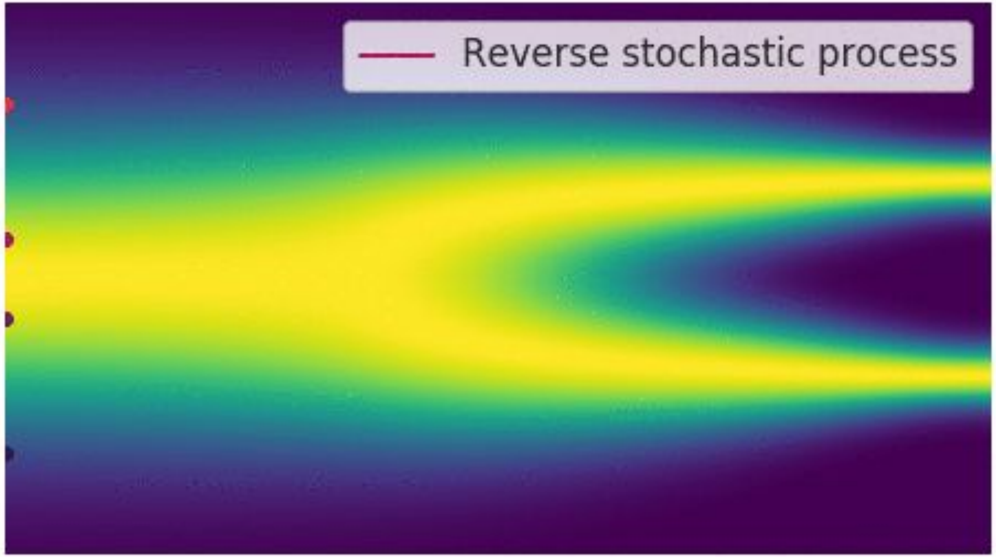


Bimodal distribution progressively diffused to Gaussian distribution

Song et al. 'Score-Based Generative Modeling through Stochastic Differential Equations' (2021)

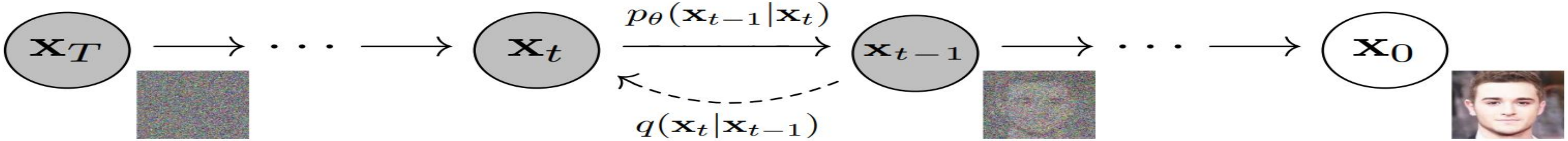
Introduction: Denoising Mechanism





Song et al. 'Score-Based Generative Modeling through Stochastic Differential Equations' (2021)

Discrete-Time



Sohl-Dickstein et al. 'Deep Unsupervised Learning using Nonequilibrium Thermodynamics', ICML 2015

- **Noising process**

$$q(x_{0:K}) = \underbrace{q_0(x_0)}_{\text{data dist}} \prod_{k=0}^{K-1} \underbrace{q(x_{k+1}|x_k)}_{\mathcal{N}(x_{k+1};\sqrt{1-\beta}x_k,\beta I)}$$

so $q(x_k|x_0) = \mathcal{N}(x_k; \sqrt{\alpha_k}x_0, (1-\alpha_k)I)$ with $\alpha_k = (1-\beta)^k$ (Ho et al., 2020).

- **Denoising process** via **Ancestral Sampling**

$$q(x_{0:K}) = \underbrace{q_K(x_K)}_{\approx\text{Gaussian}} \prod_{k=0}^{K-1} q(x_k|x_{k+1})$$

where Bayes' rule yields

$$q(x_k|x_{k+1}) = q(x_{k+1}|x_k) \overbrace{\frac{q_k(x_k)}{q_{k+1}(x_{k+1})}}^{\textbf{intractable}} \approx p_\theta(x_k|x_{k+1})$$

Discrete-Time

- **Backward transition**

$$
\begin{aligned}
q(x_k|x_{k+1}) &\approx q(x_{k+1}|x_k)\exp[\nabla \log q_k(x_{k+1})^{\mathrm{T}}(x_k - x_{k+1})] \quad \text{(Taylor)} \\
&\approx \mathcal{N}\left(x_k; \frac{1}{\sqrt{1-\beta}}\left(x_{k+1} + \beta \underbrace{\nabla \log q_k(x_{k+1})}_{\textbf{intractable}}\right), \beta I\right)
\end{aligned}
$$

- **Denoising Score matching** (Vincent (2011))

$$
\nabla \log q_k(x_k) = \mathbb{E}_{q(x_0|x_k)}[\nabla \log q(x_k|x_0)] = \frac{\sqrt{\alpha_k}\ \overbrace{\mathbb{E}[x_0|x_k]}^{\textbf{denoiser}} - x_k}{1-\alpha_k}
$$

so $\nabla \log q_k(\cdot) \approx s_{\theta^*}(k, \cdot)$ for

$$
L(\theta) = \sum_k \lambda_k \mathbb{E}_{q(x_0, x_k)}[||s_\theta(k, x_k) - \nabla \log q(x_k|x_0)||^2]
$$

Discrete-Time

Public

- **Noising mechanism**: we have for $\epsilon \sim \mathcal{N}(0, I)$

$$x_k = \sqrt{\alpha_k} x_0 + \sqrt{1-\alpha_k}\epsilon, \quad \text{so } \nabla \log q(x_k|x_0) = -\frac{\epsilon}{\sqrt{1-\alpha_k}}.$$

- **Noise parameterization** of score estimate

$$s_\theta(k, x_k) = -\frac{\epsilon_\theta(k, x_k)}{\sqrt{1-\alpha_k}} = -\frac{\epsilon_\theta(k, \sqrt{\alpha_k} x_0 + \sqrt{1-\alpha_k}\epsilon)}{\sqrt{1-\alpha_k}}.$$

- **Simple loss** (Ho et al., 2020)

$$L(\theta) = \sum_k \mathbb{E}_{x_0 \sim q_0, \epsilon \sim \mathcal{N}(0,I)}[||\epsilon_\theta(k, \sqrt{\alpha_k} x_0 + \sqrt{1-\alpha_k}\,\epsilon) - \epsilon||^2].$$

- **Generative model**: sample $x_K \sim \mathcal{N}(0, I)$ then for $k = K-1, \ldots, 0$

$$x_k = \frac{1}{\sqrt{1-\beta}}\Big(x_{k+1} + \beta s_{\theta^*}(k+1, x_{k+1})\Big) + \sqrt{\beta} \underbrace{\epsilon_k}_{\sim \mathcal{N}(0,I)}$$

From Discrete to Continuous Time

Public

- **Noising**: consider the diffusion $(x_t)_{t\in[0,T]}$

$$dx_t = -\gamma x_t dt + \sqrt{2\gamma}dw_t, \quad x_0 \sim q_0.$$

Discretized process $(x_{k\delta})_{k=0}^{K}$ for $T = K\delta$ satisfies for $\beta_\delta = 1 - \exp(-2\gamma\delta)$

$$x_{(k+1)\delta} = \sqrt{1-\beta_\delta}\, x_{k\delta} + \sqrt{\beta_\delta}\, \epsilon_{k\delta}, \quad x_0 \sim q_0.$$

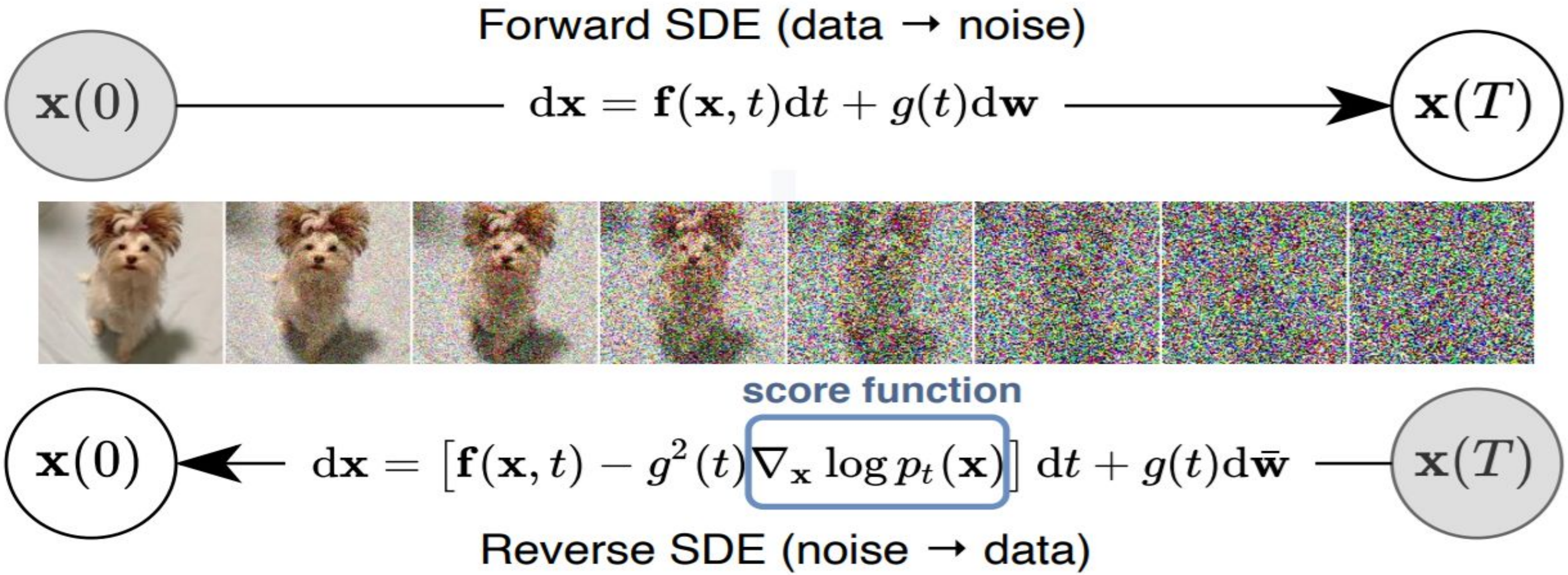
- **Denoising**: time-reversal of $(x_t)_{t\in[0,T]}$ verifies (Anderson, 1982)

$$dx_t = -\gamma(x_t + 2\nabla \log q_t(x_t))dt + \sqrt{2\gamma}d\bar{w}_t, \quad x_T \sim q_T.$$

Discretized process $(x_{k\delta})_{k=0}^{K}$ satisfies ***approximately***

$$x_{k\delta} = \frac{1}{\sqrt{1-\beta_\delta}}(x_{(k+1)\delta} + \beta_\delta \nabla \log q_{k\delta}(x_{(k+1)\delta})) + \sqrt{\beta_\delta}\bar{\epsilon}_{k\delta}, \quad x_T \sim q_T.$$

From Discrete To Continuous-Time



Song et al. 'Score-Based Generative Modeling through Stochastic Differential Equations' ICLR 2021

Learning the Score

- **Time-reversal**:

$$\mathrm{d}x_t = -\gamma(x_t + 2\nabla \log q_t(x_t))\mathrm{d}t + \sqrt{2\gamma}\mathrm{d}\bar{w}_t, \quad x_T \sim q_T$$

  induces path measure $\mathcal{Q}$.

- **Approximate time-reversal**:

$$\mathrm{d}x_t = -\gamma(x_t + 2s_\theta(t, x_t))\mathrm{d}t + \sqrt{2\gamma}\mathrm{d}\bar{w}_t, \quad x_T \sim \mathcal{N}(0, I)$$

  induces path measure $\mathcal{P}_\theta$.

- **Learning the score**: Minimizing $\mathrm{KL}(\mathcal{Q}||\mathcal{P}_\theta)$ w.r.t. θ is equivalent to minimizing

$$\mathcal{L}(\theta) = \int_0^T \mathbb{E}_{x_0, x_t \sim q_{0,t}}[||s_\theta(t, x_t) - \nabla \log q_{t|0}(x_t|x_0)||^2]\mathrm{d}t$$

Theoretical Results

Convergence of diffusion models (De Bortoli et al., 2021)

- Assume there exists $M \geq 0$ such that for any $t \in [0, T]$ and $x \in \mathbb{R}^d$

$$||\mathbf{s}_{\theta^\star}(t, x) - \nabla \log p_t(x)|| \leq M ,$$

with $\mathbf{s}_{\theta^\star} \in C([0, T] \times \mathbb{R}^d, \mathbb{R}^d)$ and regularity conditions on the density of π w.r.t. the Lebesgue measure and its gradients.

- Then there exist $B, C, D \geq 0$ s.t. for any $N \in \mathbb{N}$ and $\{\gamma_k\}_{k=1}^N$ the following hold:

$$\boxed{\|\mathcal{L}(Y_N) - \pi\|_{\mathrm{TV}} \leq B \exp[-T] + C(M + \gamma^{1/2}) \exp[DT] \; .}$$

where $T = N\gamma$.

- **A few remarks**:
  - The assumption on π is *not* satisfied if π defined on a **manifold** of $\mathbb{R}^d$ with dimension $p < d$, see De Bortoli (2022)
  - The approximation assumption is strong and could be **relaxed**.
  - The term $\exp[DT]$ can be improved to a **polynomial dependency**.
  - Extension & Improvements Lee et al. (2022); Chen et al. (2022, 2023).

Benefits of Continuous-Time

- **Sophisticated numerical integrators**, predictor-corrector schemes.
- **Deterministic sampling & Likelihood** (Song et al., 2021):
  Probability flow ODE

$$\mathrm{d}x_t = \underbrace{-\gamma(x_t + \nabla \log q_t(x_t))}_{f(t, x_t)} \mathrm{d}t, \qquad x_0 \sim q_0$$

  s.t. $\mathrm{Law}(x_t) = q_t$ so **deterministic generation** and **likelihood computation** possible.
- **Approximate Likelihood computation**

$$\log q_0(x_0) = \log q_T(x_T) + \int_0^T \nabla \cdot f(t, x_t) \mathrm{d}t.$$

Bayesian Inference

Public

- **Posterior distribution**

$$p_0(x_0) \propto \underbrace{q_0(x_0)}_{\text{can be sampled using DDM}} \underbrace{p(y|x_0)}_{\text{likelihood}}$$

- **DDMs Sampling** uses $\nabla \log p_t(x_t)$ for $p_t(x_t) = \int p_0(x_0) q_{t|0}(x_t|x_0) \mathrm{d}x_0$.
  - **Amortization** The identity

$$\nabla \log p_t(x_t) = \int \nabla \log q_{t|0}(x_t|x_0) p(x_0|x_t, y) \mathrm{d}x_0$$

shows that $\nabla \log p_t(x_t) \approx s_{\theta^*}(t, x_t, y)$ for θ^* minimizing

$$L(\theta) = \mathbb{E}_{x_0, y, x_t}[||s_\theta(t, x_t, y) - \nabla \log q_{t|0}(x_t|x_0)||^2]$$

  - **Guidance**: Alternatively, we can use

$$\nabla \log p_t(x_t) = \nabla \log q_t(x_t) + \nabla \log p_t(y|x_t)$$

for $p_t(y|x_t) = \int p(y|x_0) q_{0|t}(x_0|x_t) \mathrm{d}x_0$.

Image Super-Resolution

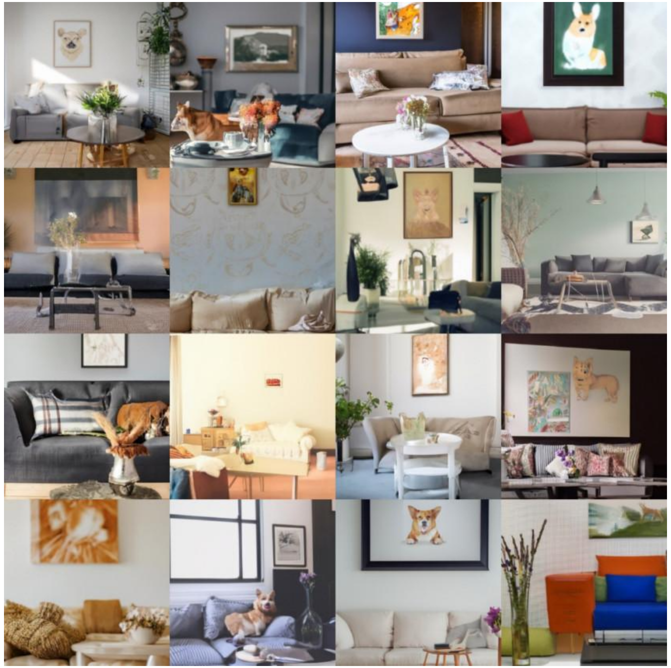
- Posterior samples for CelebA super-resolution task with noise for $N = 20$ ($d = 116, 412$). From left to right: ground thruth, y^{obs}, eight posterior samples and their mean.

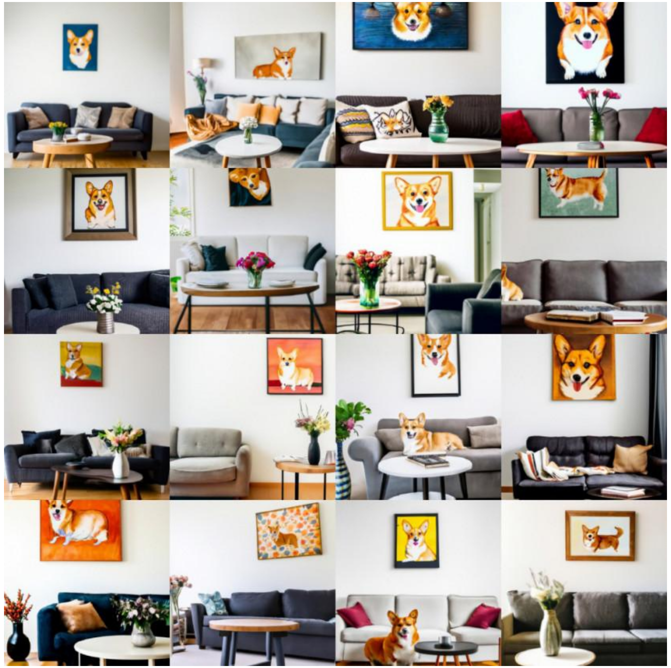
Y. Shi, V. De Bortoli, G. Deligiannidis & A.D., UAI 2022

Text-to-Image: The power of classifier-free guidance

A cozy living room with a painting of a corgi on the wall […]

Public

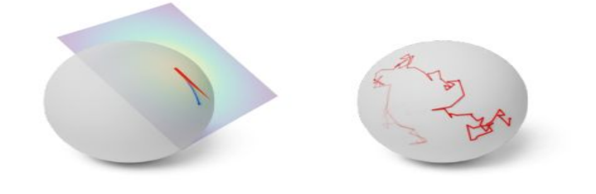




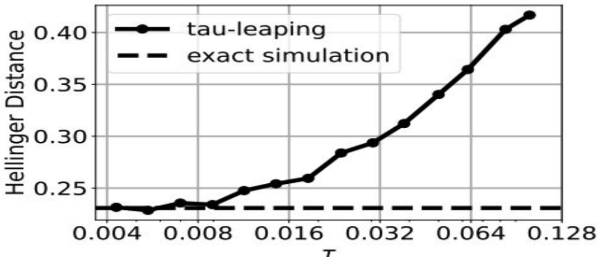
Nichol et al. 'GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models' (2021)
https://benanne.github.io/2022/05/26/guidance.html

Extensions Beyond $\mathbb{R}^d$

- **Riemmanian manifolds** (De Bortoli et al., 2022; Huang et al., 2022): applications to protein backbone generation (Watson et al., 2022) and joint grasp and motion optimization (Urain et al., 2022).
- **Discrete state-space** (Campbell et al., 2022): allows to exploit fast samplers from chemical physics.
- **Simplex** (Benton et al., 2022; Richemond et al., 2022): appplication to compositional data.
- **Function spaces** (Dutordoir et al., 2022; Kerrigan et al., 2022)



(a) A single step of a Geodesic Random Walk. (b) Many steps yield an approximate trajectory.



(a) GP Regression (b) Attentive Latent NP (c) Neural Diffusion Process (ours)

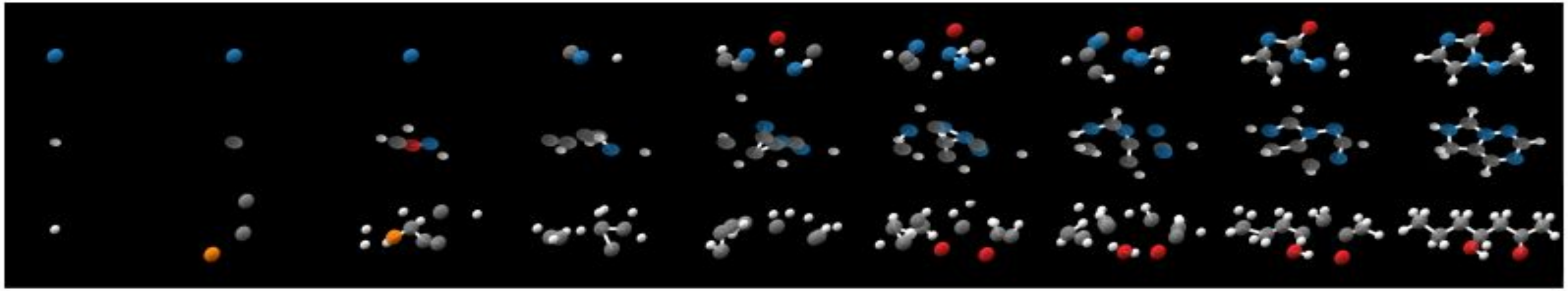
Dutortoir et al., "Neural Diffusion Processes"

Trans-dimensional generative modeling via jump diffusions

- **Data of varying dimension**: molecules with varying number of atoms, videos with varying number of frames; i.e. π lives on $\cup_{k=1}^{k_{\max}} \mathbb{R}^k$.
- **Difficulties**: Training $k_{\max}$ models expensive and conditional generation would require predicting k given observations.
- **Jump Diffusion**: We diffuse and kill components until one remains and is Gaussian, time-reversal adds components.

Algorithm 1: Sampling the Generative Process

$t \leftarrow T$
$\mathbf{X} \sim p_{\text{ref}}(\mathbf{X}) = \mathbb{I}\{n=1\}\mathcal{N}(\mathbf{x}; 0, I_d)$
while $t > 0$ **do**
  if $u < \overleftarrow{\lambda}_t^{\theta}(\mathbf{X})\delta t$ with $u \sim \mathcal{U}(0,1)$ **then**
    Sample $\mathbf{x}^{\text{add}}, i \sim A_t^{\theta}(\mathbf{x}^{\text{add}}, i|\mathbf{X})$
    $\mathbf{X} \leftarrow \text{ins}(\mathbf{X}, \mathbf{x}^{\text{add}}, i)$
  end
  $\mathbf{x} \leftarrow \mathbf{x} - \overleftarrow{\mathbf{b}}_t^{\theta}(\mathbf{X})\delta t + g_t\sqrt{\delta t}\epsilon$ with $\epsilon \sim \mathcal{N}(0, I_{nd})$
  $\mathbf{X} \leftarrow (n, \mathbf{x}), t \leftarrow t - \delta t$
end



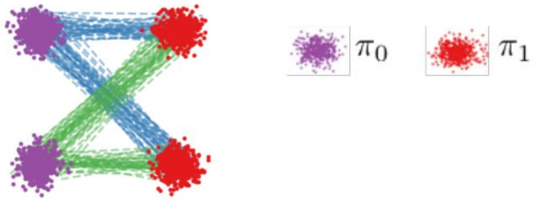
Visualization of the jump-diffusion backward generative process on molecules.

Campbell et al., arXiv:2305.16261

Public

Beyond Diffusion Models: Transport using ODEs

- **Diffusion models**: noising "transports" q_0 to $q_T \approx \mathcal{N}(0, I)$, denoising does the reverse.
- **Limitations**: What if T is not large enough? What if you want q_T be non-normal? Do we need diffusions at all?
- **Interpolants** (Liu et al., 2022, 2023; Lipman et al., 2023; Albergo and Vanden-Eijnden, 2023): Let $x_0 \sim q_0$ and $x_1 \sim q_1$ and define $x_t = (1-t)\,x_0 + t\,x_1 \sim q_t$.



(a) Linear interpolation
$X_t = tX_1 + (1-t)X_0$

Liu et al.., "Flow Straight and Fast", ICLR 2023

Beyond Diffusion Models: Transport using ODEs

Public

- **Useless Transport ODE**: As $x_t = x_0 + t(x_1 - x_0)$ then the ODE

$$\frac{\mathrm{d}x_t}{\mathrm{d}t} = x_1 - x_0, \qquad x_0 \sim q_0,\ x_1 \sim q_1,$$

has marginals $(q_t)_{t\in[0,1]}$. Useless for generative modeling!

- **Useful Transport ODE** (Liu et al. (2023)): The ODE with drift

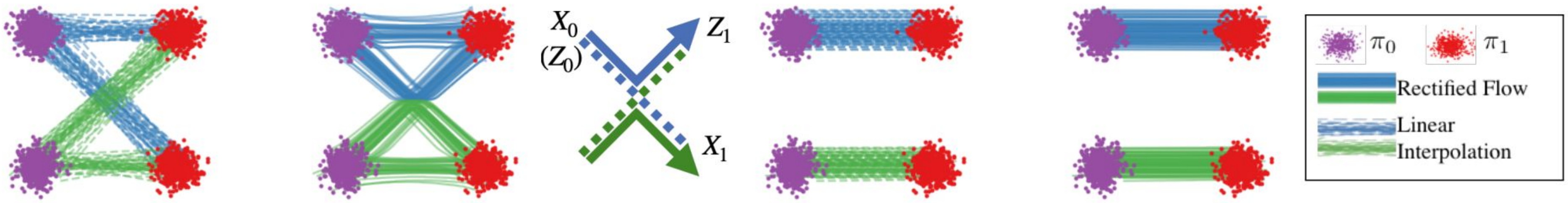
$$v(t,x) = \mathbb{E}_{q(x_0,x_1|x_t)}[x_1 - x_0 | x_t = x]$$

admits the same marginals $(q_t)_{t\in[0,T]}$!

- **Learning the Drift**: Learn $u_{\theta^\star}(t,\cdot) \approx v(t,\cdot)$ by minimizing

$$L(\theta) = \mathbb{E}_{q(x_0,x_1)}[||u_\theta(t,(1-t)x_0 + tx_1) - (x_1 - x_0)||^2]$$

Beyond Diffusion Models: Transport using ODEs



(a) Linear interpolation $X_t = tX_1 + (1-t)X_0$

(b) Rectified flow Z_t induced by (X_0, X_1)

(c) Linear interpolation $Z_t = tZ_1 + (1-t)Z_0$

(d) Rectified flow Z'_t induced by (Z_0, Z_1)

Figure 2: (a) Linear interpolation of data input $(X_0, X_1) \sim \pi_0 \times \pi_1$. (b) The rectified flow Z_t induced by (X_0, X_1); the trajectories are "rewired" at the intersection points to avoid the crossing. (c) The linear interpolation of the end points (Z_0, Z_1) of flow Z_t. (d) The rectified flow induced from (Z_0, Z_1), which follows straight paths.

Liu et al., "Flow Straight and Fast", ICLR 2023

Diffusion Schrödinger Bridge Matching

- **Schrödinger Bridge**: Find

$$\Pi^* = \arg\min_{\Pi}\Big\{\mathrm{KL}(\Pi|\mathcal{Q}) : \Pi_0 = q_0, \Pi_1 = q_1\Big\}.$$

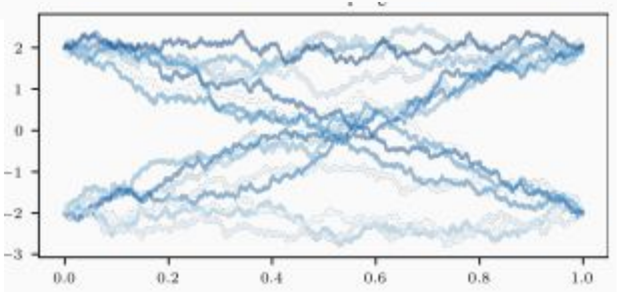
where $\mathcal{P}$ is a Markov reference measure, i.e. Brownian.

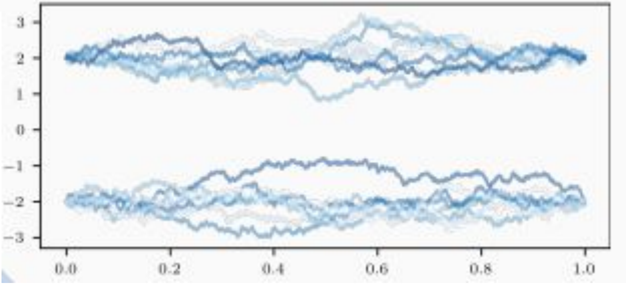
- The SB satisfies $\Pi^* = \pi^s \mathcal{Q}_{|0,T}$ for π^s a static Schrödinger bridge. For $\mathcal{Q}$ Brownian, Π^* is entropy-regularized OT.
- Standard method is **Iterative Proportional Fitting**: let $\Pi^0 = \mathcal{Q}$ then

$$\Pi^{2n+1} = \arg\min_{\Pi}\Big\{\mathrm{KL}(\Pi|\Pi^{2n}) : \Pi_1 = q_1\Big\},$$
$$\Pi^{2n+2} = \arg\min_{\Pi}\Big\{\mathrm{KL}(\Pi|\Pi^{2n+1}) : \Pi_0 = q_0\Big\},$$

Diffusion Schrödinger Bridge Matching

- **Diffusion Schrödinger Bridge Matching**: Alternate between Markovian projection and reciprocal projection

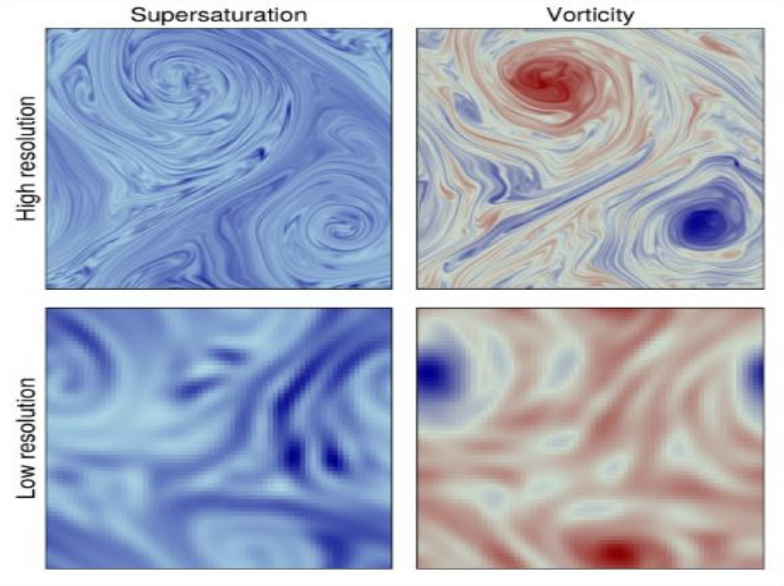




Y. Shi, V. De Bortoli, A. Campbell & A.D. arXiv:2303.16852
S. Peluchetti, arXiv:2304.00917

Diffusion Schrödinger Bridge Matching

- Application in **climate science**:
  - **Downscaling**: high resolution data from low resolution ones.
  - This is a **super resolution** task.
  - No **paired datasets** of high and low resolutions exist.



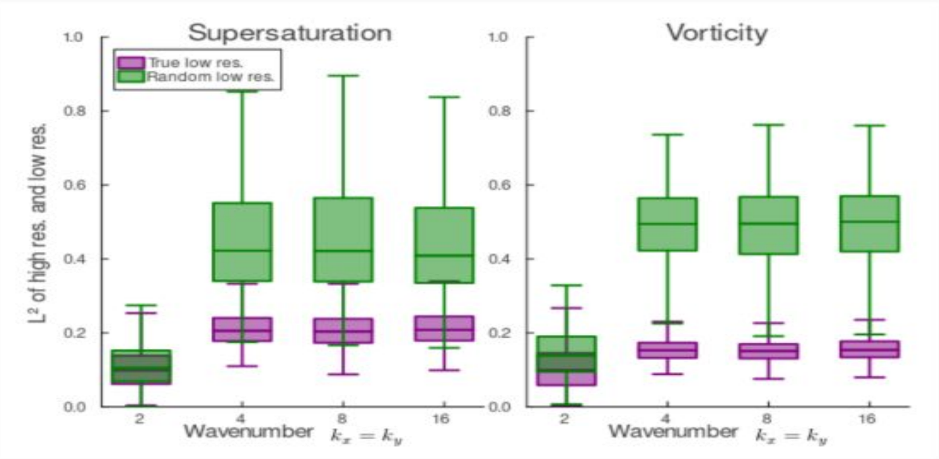
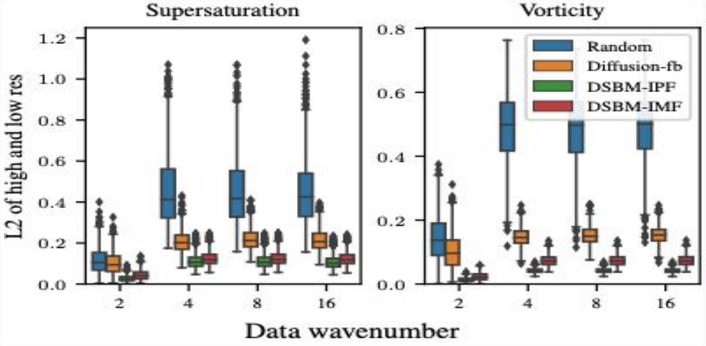
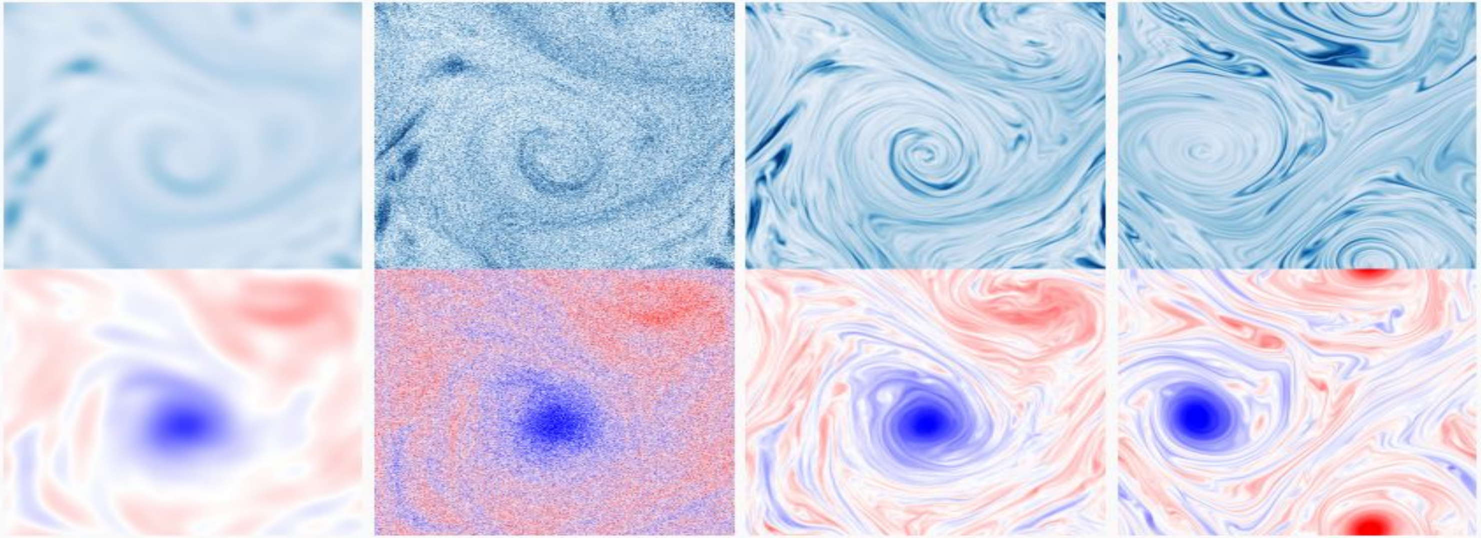


Image extracted from Bischoff and Deck (2023).

Diffusion Schrödinger Bridge Matching

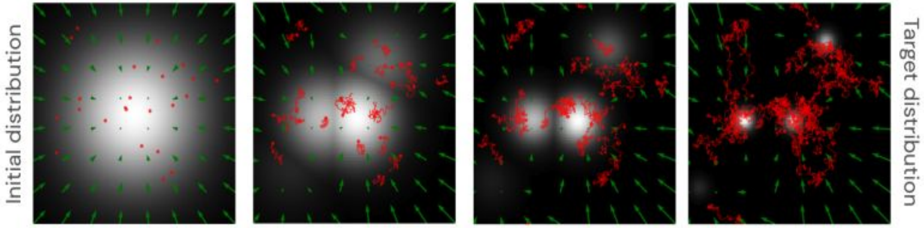
- Same setting as Bischoff and Deck (2023).
- **Super resolution** task.
- Quality measure (frequency histogram).
- Similarity measure (ℓ_2 with upscaling).





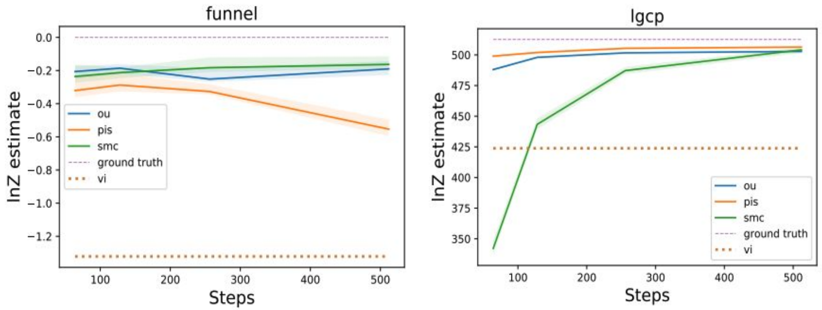
Beyond Generative Modeling: Integration & Sampling

$$X_0 \sim \pi_0 \longrightarrow \mathrm{d}X_t = \nabla \log \pi_t(X_t)\mathrm{d}t + \sqrt{2}\mathrm{d}B_t \longrightarrow X_T \approx \pi$$



$$\bar{X}_T \approx \pi_0 \longleftarrow \mathrm{d}\bar{X}_t = \{-\nabla \log \pi_{T-t}(\bar{X}_t) + 2\nabla \log q_{T-t}(\bar{X}_t)\}\,\mathrm{d}t + \sqrt{2}\mathrm{d}\bar{B}_t \longleftarrow \bar{X}_0 \sim \pi$$

D., Grathwohl et al., "Score-based diffusion meets AIS", NeurIPS 2022



Vargas et al., "Denoising diffusion samplers", ICLR 2023

- **Normalizing Constant**: Annealed Importance Sampling is gold standard but can be improved using score ideas (Doucet et al., 2022).
- **Sampling Unnormalized Targets**: Target is "diffused" to Gaussian (Vargas et al., 2023) or delta-mass (Zhang and Chen, 2022), time-reversal/scores learned using reverse KL.
- **Simulation SDEs**: Given $\mathrm{d}x_t = f(t, x_t)\mathrm{d}t + \mathrm{d}w_t$, probability flow ODE is $\mathrm{d}x_t = (f(t, x_t) - \frac{1}{2}\nabla \log q_t(x_t))\mathrm{d}t$ and estimate sequentially score (Boffi and Vanden-Eijnden, 2022).

Discussion

Public

- **Denoising Diffusion Models** provide state-of-the-art performance in numerous domains: image, audio, proteins etc.
- **Dynamic transport** alternatives are now also available.
- Available **theoretical results** do not explain the remarkable empirical performance: **Exact score is useless for generative modeling!**
- **Exciting research area** at the interface of applied maths/proba, machine learning, Monte Carlo and statistics.

References

Public

Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. *ICLR*, 2023.

Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

Joe Benton, Yuyang Shi, Valentin De Bortoli, George Deligiannidis, and Arnaud Doucet. From denoising diffusions to denoising markov models. *arXiv preprint arXiv:2211.03595*, 2022.

Nicholas M Boffi and Eric Vanden-Eijnden. Probability flow solution of the fokker-planck equation. *arXiv preprint arXiv:2206.04642*, 2022.

Andrew Campbell, Joe Benton, Valentin De Bortoli, Tom Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. *NeurIPS*, 2022.

References

Valentin De Bortoli, Emile Mathieu, Michael Hutchinson, James Thornton, Yee Whye Teh, and Arnaud Doucet. Riemannian score-based generative modeling. *NeurIPS*, 2022.

Arnaud Doucet, Will Grathwohl, Alexander GDG Matthews, and Heiko Strathmann. Score-based diffusion meets annealed importance sampling. *NeurIPS*, 2022.

Vincent Dutordoir, Alan Saul, Zoubin Ghahramani, and Fergus Simpson. Neural diffusion processes. *arXiv preprint arXiv:2206.03992*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Chin-Wei Huang, Milad Aghajohari, Avishek Joey Bose, Prakash Panangaden, and Aaron Courville. Riemannian diffusion models. *NeurIPS*, 2022.

Gavin Kerrigan, Justin Ley, and Padhraic Smyth. Diffusion generative models in infinite dimensions. *arXiv preprint arXiv:2212.00886*, 2022.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *ICLR*, 2023.

References

Xingchao Liu, Lemeng Wu, Mao Ye, and Qiang Liu. Let us build bridges: Understanding and extending diffusion generative models. *NeurIPS Workshop SBM*, 2022.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *ICLR*, 2023.

Pierre H Richemond, Sander Dieleman, and Arnaud Doucet. Categorical SDEs with simplex diffusion. *arXiv preprint arXiv:2210.14784*, 2022.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *ICLR*, 2021.

Julen Urain, Niklas Funk, Georgia Chalvatzaki, and Jan Peters. Se (3)-diffusionfields: Learning cost functions for joint grasp and motion optimization through diffusion. *arXiv preprint arXiv:2209.03855*, 2022.

Francisco Vargas, Will Grathwohl, and Arnaud Doucet. Denoising diffusion samplers. *ICLR*, 2023.

References

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

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. Broadly applicable and accurate protein design by integrating structure prediction networks and diffusion generative models. *bioRxiv*, 2022.

Qinsheng Zhang and Yongxin Chen. Path integral sampler: a stochastic control approach for sampling. *ICLR*, 2022.

More convergence

Public

- The central decomposition

$$
\begin{aligned}
||\mathcal{L}(X_0) - p_{\text{data}}||_{\text{TV}} &= ||p_{\text{prior}}\hat{\mathrm{R}}_N - p_{\text{data}}||_{\text{TV}} \\
&= ||p_{\text{prior}}\hat{\mathrm{R}}_N - p_T\mathrm{Q}_T||_{\text{TV}} \\
&\leq ||p_{\text{prior}}\hat{\mathrm{R}}_N - p_{\text{prior}}\mathrm{Q}_T||_{\text{TV}} + ||p_T\mathrm{Q}_T - p_{\text{prior}}\mathrm{Q}_T||_{\text{TV}} \\
&\leq ||p_{\text{prior}}\hat{\mathrm{R}}_N - p_{\text{prior}}\mathrm{Q}_T||_{\text{TV}} + ||p_{\text{data}}\mathrm{P}_T - p_{\text{prior}}||_{\text{TV}},
\end{aligned}
$$

  where
  - $(\mathrm{P}_t)_{t\geq 0}$ is the **forward** Ornstein-Ulhenbeck semi-group,
  - $(\mathrm{Q}_t)_{t\geq 0}$ is the **backward** Ornstein-Ulhenbeck semi-group,
  - $(\hat{\mathrm{R}}_n)_{n\in\{1,\ldots,N\}}$ is the iterated kernel associated with the backward Markov chain.
- $||p_{\text{prior}}\hat{\mathrm{R}}_N - p_{\text{prior}}\mathrm{Q}_T||_{\text{TV}}$: **approximation error** $\rightarrow$ Girsanov theorem.
- $||p_{\text{data}}\mathrm{P}_T - p_{\text{prior}}||_{\text{TV}}$: **geometric ergodicity** of Ornstein-Ulhenbeck.