

Topics in SVGD and Diffusion Model

ZHIJUN ZENG

2023年5月28日

目录

1	SVGD	1
1.1	Notation	1
1.2	Stein's Identity and Kernelized Stein Discrepancy	1
1.3	Variational Inference Using Smooth Transforms	?
1.4	Stein Operator as Derivative of KL divergence	?
1.5	Algorithm	?
2	Kernel Stein Generative Modeling	?
3	Score Matching=Conditional Score Matching	?
3.1	Recap of Score Based Diffusion Model	?
3.2	Score Matching is Conditional Score Matching	?
3.3	Inequality	?
3.4	Equivalence	?
4	Recap of Poisson Flow Model	1
4.1	Forward sampling	?

5	GENPHYS	?
5.1	Generative Model from Physical Process	1
5.1.1	GENERATIVE MODELS AS DENSITY FLOW	?
5.1.2	PHYSICAL PROCESSES AND PHYSICAL PDES	?
5.1.3	Converting Physical PDE to Density Flows	?
5.1.4	Diffusion model	?
5.1.5	Poisson Flow Generative Models	?
5.2	WHICH PHYSICAL PROCESSES CAN BE GENPHYS	?
5.2.1	Ideal wave equation (not s -generative)	?
5.2.2	Dissipative wave equation (conditionally s -generative)	?
5.2.3	Helmholtz(conditionally s -generative)	?
5.2.4	Screened Poisson Equation (s – generative)	?
5.3	Dispersion Relation as A Criterion	?
6	Consistency Model	?
6.1	Recap of Score Based Diffusion Model	?
6.2	Consistency Models	?
6.2.1	Sampling	?
6.3	Training Consistency Models via Distillation	?
6.4	Training Consistency Models in Isolation	?

1 SVGD

1.1 Notation

x : Random variable taking value on $\chi \subset \mathbb{R}^d$ with i.i.d observation $\{D_k\}$

$p_0(x)$: Prior distribution

posterior: $p(x) = \frac{p_0(x) \prod_{k=1}^N p_0(D_k|x)}{\int \bar{p}(x) dx}$

定义 1. Let $k(x, x'): \chi \times \chi \rightarrow \mathbb{R}$ be a positive kernel, the RKHS \mathcal{H} of $k(x, x')$ is the closure of

$$\left\{ f: f(x) = \sum_{i=1}^m a_i k(x, x_i), a_i \in \mathbb{R}, m \in \mathbb{N}, x_i \in \chi \right\}$$

with an inner product $\langle f, g \rangle = \sum_{i,j} a_i b_j k(x_i, x_j)$. $\mathcal{H}^d = \Pi_{i=1}^d \mathcal{H}$ with $\langle f, g \rangle_{\mathcal{H}^d} = \sum_{i=1}^d \langle f_i, g_i \rangle$

1.2 Stein's Identity and Kernelized Stein Discrepancy

Let $p(x)$ be smooth density supported on $\chi \subset \mathbb{R}^d$, $\phi(x) = [\phi^1(x), \dots, \phi^d(x)]^T$ be smooth function.

命题 2. For sufficient regular ϕ , here $\mathcal{A}_p\phi(x)$ is a matrix

$$\mathbb{E}_{x \sim p}[\mathcal{A}_p\phi(x)] = 0, \quad \text{where} \quad \mathcal{A}_p\phi(x) = \phi(x) \nabla_x \log p(x)^\top + \nabla_x \phi(x), \quad (1)$$

here \mathcal{A}_p is called stein operator. The set ϕ of such p is called Stein Class of p .

Let q be a smooth density supported on $\chi \subset \mathbb{R}^d$, then $\mathbb{E}_{x \sim q}[\mathcal{A}_p\phi(x)]$ relates to how different p and q are.

定义 3. Stein discrepancy is the maximum violation of Stein's identity for ϕ in some function set \mathcal{F}

$$\mathbb{S}(q, p) = \max_{\phi \in \mathcal{F}} \{ \mathbb{E}_{x \sim q} [\text{Tr} \mathcal{A}_p\phi(x)]^2 \}$$

The Kernelized Stein Discrepancy is to maximizing the Stein discrepancy in the unit ball of a reproducing kernel Hilbert space

定义 4. Given (p, q) , Kernelized Stein Discrepancy

$$\mathbb{S}(q, p) = \max_{\phi \in \mathcal{H}^d} \{ \mathbb{E}_{x \sim q} [\text{Tr} \mathcal{A}_p \phi(x)]^2, \|\phi\|_{\mathcal{H}^d} \leq 1 \}$$

Here we assume that $k(x, x')$ is in Stein class fixed x' . The optimal solution of KSD is $\phi(x) = \frac{\phi_{q,p}^*(x)}{\|\phi_{q,p}^*\|_{\mathcal{H}^d}}$

$$\phi_{q,p}^*(x) = \mathbb{E}_{x' \sim q} [\mathcal{A}_p k(x', x)], \mathbb{S}(q, p) = \|\phi_{q,p}^*\|_{\mathcal{H}^d}$$

Further we can show that $\mathbb{S}(q, p) = 0$ iff $p = q$ and $k(x, x')$ strictly positive. Commonly we choose RBF $k(x, x') = \exp(-\frac{1}{h}\|x - x'\|_2^2)$.

Another observation is that KSD and Stein operator depend on p only through the score function $\nabla_x \log p(x)$ which can be calculated without knowing the normalization constant of p .

1.3 Variational Inference Using Smooth Transforms

Variational inference approximates the target distribution $p(x)$ using a simpler distribution $q^*(x)$ found in a predefined set $Q = \{q(x)\}$ of distributions by minimizing the KL divergence,

$$q^* = \operatorname{argmin}\{\operatorname{KL}(q||p) \equiv \mathbb{E}_q[\log q(x)] - \mathbb{E}_q[\log \bar{p}(x)] + \log Z\},$$

Here we consider Q as the the set of distributions of random variables of form $z = \mathbf{T}(x)$ where $\mathbf{T}: \chi \rightarrow \chi$ is a smooth one-to-one transform, and x is drawn from a tractable reference distribution $q_0(x)$. The change of variables formula says

$$q_{[\mathbf{T}]}(z) = q(\mathbf{T}^{-1}(z)) \cdot |\det (\nabla_z \mathbf{T}^{-1}(z))|,$$

1.4 Stein Operator as Derivative of KL divergence

We consider an incremental transform formed by a small perturbation of the identity map: $\mathbf{T}(x) = x + \varepsilon \phi(x)$, then it is one-to-one.

定理 5. $\mathbf{T}(x) = x + \varepsilon \phi(x)$, $q_{[\mathbf{T}]}(z)$ is the density of $\mathbf{z} = \mathbf{T}(\mathbf{x})$, $x \sim q(x)$

$$\nabla_{\varepsilon} \text{KL}(q_{[\mathbf{T}]} \parallel p) \big|_{\varepsilon=0} = -\mathbb{E}_{x \sim q}[\text{Tr}(\mathcal{A}_p \phi(x))],$$

where $\mathcal{A}_p \phi(x) = \phi(x) \nabla_x \log p(x)^{\top} + \nabla_x \phi(x)$.

In zero-centered balls of \mathcal{H}^d , the steepest descent direction is $\phi_{q,p}^*$

引理 6. Consider all the perturbation directions ϕ in the ball $\mathcal{B} = \{\phi \in \mathcal{H}^d: \|\phi\|_{\mathcal{H}^d} \leq \mathbb{S}(q, p)\}$, the direction of steepest descent that maximizes the negative gradient is $\phi_{q,p}^*$

$$\phi_{q,p}^*(x) = \mathbb{E}_{x' \sim q}[\mathcal{A}_p k(x', x)] = \mathbb{E}_{x' \sim q}[k(x', x) \nabla_{x'} \log p(x') + \nabla_{x'} k(x', x)]$$

$$\nabla_{\varepsilon} \text{KL}(q_{[T]} || p) |_{\varepsilon=0} = -\mathbb{S}(q, p).$$

By this lemma , we can construct a iterative procedure that transforms an distribution q_0 to the target distribution p : From q_i , applying transform $T_i^*(x) = x + \varepsilon_i \phi_{q_i,p}^*(x)$, which decreases the KL divergence by $\varepsilon_i \mathbb{S}(q_i, p)$, doing this iteratively.

定理 7. $T(x) = x + f(x)$, where $f \in \mathcal{H}^d, x \sim q$

$$\nabla_f \text{KL}(q_{[T]} || p) |_{f=0} = -\phi_{q,p}^*$$

whose squared RKHS norm is $\|\phi_{q,p}^*\|_{\mathcal{H}^d}^2 = \mathbb{S}(q, p)$.

This suggests that $T^*(x) = x + \varepsilon \phi_{q,p}^*(x)$ is equivalent to a step of functional gradient descent in RKHS.

1.5 Algorithm

Algorithm 1 Bayesian Inference via Variational Gradient Descent

Input: A target distribution with density function $p(x)$ and a set of initial particles $\{x_i^0\}_{i=1}^n$.

Output: A set of particles $\{x_i\}_{i=1}^n$ that approximates the target distribution.

for iteration ℓ **do**

$$x_i^{\ell+1} \leftarrow x_i^\ell + \epsilon_\ell \hat{\phi}^*(x_i^\ell) \quad \text{where} \quad \hat{\phi}^*(x) = \frac{1}{n} \sum_{j=1}^n [k(x_j^\ell, x) \nabla_{x_j^\ell} \log p(x_j^\ell) + \nabla_{x_j^\ell} k(x_j^\ell, x)], \quad (8)$$

where ϵ_ℓ is the step size at the ℓ -th iteration.

end for

The SDE form of SVGD is

$$\frac{d\mathbf{x}_t}{dt} = \mathbb{E}_{\mathbf{x} \sim p_t(\mathbf{x})} [k(\mathbf{x}, \mathbf{x}_t) \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \nabla_{\mathbf{x}} k(\mathbf{x}, \mathbf{x}_t)]$$

2 Kernel Stein Generative Modeling

Here introduce a deep noise conditional kernel(NCK):

$$k_{\psi}(\mathbf{x}, \mathbf{x}'; \sigma) = k_{\text{rbf}}(E_{\psi}(\mathbf{x}, \sigma), E_{\psi}(\mathbf{x}', \sigma); \sigma) + k_{\text{img}}(E_{\psi}(\mathbf{x}, \sigma), E_{\psi}(\mathbf{x}', \sigma); \sigma)$$

where $k_{\text{rbf}}(\mathbf{x}, \mathbf{x}'; \sigma) = \exp(-\gamma(\sigma) \|\mathbf{x} - \mathbf{x}'\|_2^2)$, $k_{\text{img}}(\mathbf{x}, \mathbf{x}'; \sigma) = (1 + \|\mathbf{x} - \mathbf{x}'\|_2^2)^{\tau(\sigma)}$.

The deep encoder $E_{\psi}(\mathbf{x}, \sigma): \mathbb{R}^d \rightarrow \mathbb{R}^h$ is learned by noise conditional auto-encoder

$$\frac{1}{2L} \sum_{l=1}^L \frac{1}{\sigma_l^2} \cdot \mathbb{E}_{p_{\sigma_l}(\tilde{\mathbf{x}}|\mathbf{x}) p_d(\mathbf{x})} [\|D_{\varphi}(E_{\psi}(\tilde{\mathbf{x}}, \sigma_l), \sigma_l) - \mathbf{x}\|^2]$$

where $D_{\varphi}(\mathbf{x}, \sigma_l): \mathbb{R}^h \rightarrow \mathbb{R}^d$ is the corresponding decoder.

Then we modify the learning target to

$$\min_q F_\beta(q) = \text{KL}(q, p) - (\beta - 1)H(q) = \beta \text{KL}\left(q, p^{\frac{1}{\beta}}\right)$$

命题 8. Consider continuous Stein descent

$$dX_t = \phi_{q_t, p, \beta}^*(X_t) dt, \quad \phi_{q_t, p, \beta}^*(X_t) = \mathbb{E}_{x' \sim q_t} [\nabla_{x'} \log p(x') K(x', x) + \beta \nabla_{x'} K(x', x)]$$

we have $\frac{dF_\beta(q_t)}{dt} = -\beta^2 \mathbb{S}^2\left(q_t, p^{\frac{1}{\beta}}\right)$ and $\phi_{q_t, p, \beta}^*$ is a decent direction for entropy regularized KL divergence.

Algorithm 1: NCK-SVGD with Entropic Regularization.

input : $\{\sigma_l\}_{l=1}^L$ the data-perturbing noises, $s_\theta(\mathbf{x}, \sigma)$ the noise conditional score function,
 $k_\psi(\mathbf{x}, \mathbf{x}'; \sigma)$ the noise conditional kernel, a set of initial particles $\{\mathbf{x}_i^{(0)}\}_{i=1}^n$,
the entropic regularizer β , an initial learning rate ϵ , and a maximum iteration T .

output : A set of particles $\{\mathbf{x}_i^{(T)}\}_{i=1}^n$ that approximates the target distribution.

for $l \leftarrow 1$ **to** L **do**

$\eta_l \leftarrow \epsilon \cdot (\sigma_l / \sigma_L)^2$

for $t \leftarrow 1$ **to** T **do**

$\mathbf{x}_i^t \leftarrow \mathbf{x}_i^{(t-1)} + \eta_l \cdot \phi_{q_t, p, \beta}^*(\mathbf{x}_i^{(t-1)}), \quad \text{where}$

$\phi_{q_t, p, \beta}^*(\mathbf{x}) = \frac{1}{n} \sum_{j=1}^n \left[k_\psi(\mathbf{x}_j^{(t-1)}, \mathbf{x}; \sigma_l) s_\theta(\mathbf{x}, \sigma_l) + \beta \cdot \nabla_{\mathbf{x}_j^{(t-1)}} k_\psi(\mathbf{x}_j^{(t-1)}, \mathbf{x}; \sigma_l) \right].$

$\mathbf{x}_i^{(0)} \leftarrow \mathbf{x}_i^{(T)}, \forall i = 1, \dots, n.$

3 Score Matching=Conditional Score Matching

3.1 Recap of Score Based Diffusion Model

The forward process is described by SDE:

$$d\mathbf{x}_t = \mathbf{f}_t(\mathbf{x}_t)dt + \mathbf{g}_t(\mathbf{x}_t)dW_t$$

With backward process:

$$d\mathbf{x}_t = [\mathbf{f}_t(\mathbf{x}_t) - \mathbf{g}_t(\mathbf{x}_t)^2 \nabla_{\mathbf{x}} \log(p_t(\mathbf{x}_t))]dt + \mathbf{g}_t d\hat{W}_t$$

Using Law of total expectation

$$p(\mathbf{x}_t) = \int p(\mathbf{x}_t|\mathbf{x}_0)\tilde{p}(\mathbf{x}_0)d\mathbf{x}_0 = \mathbb{E}_{\mathbf{x}_0}[p(\mathbf{x}_t|\mathbf{x}_0)]$$

Then The score function at time t is

$$\nabla_{\mathbf{x}_t} \log(p(\mathbf{x}_t)) = \frac{\mathbb{E}_{\mathbf{x}_0}[\nabla_{\mathbf{x}_t} p(\mathbf{x}_t|\mathbf{x}_0)]}{\mathbb{E}_{\mathbf{x}_0}[p(\mathbf{x}_t|\mathbf{x}_0)]} = \frac{\mathbb{E}_{\mathbf{x}_0}[p(\mathbf{x}_t|\mathbf{x}_0) \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)]}{\mathbb{E}_{\mathbf{x}_0}[p(\mathbf{x}_t|\mathbf{x}_0)]}$$

We expect to approximate score function $\nabla_{\mathbf{x}_t} \log(p(\mathbf{x}_t))$ by a Neural Network $\mathbf{s}_\theta(\mathbf{x}_t, t)$, here we design the target as minimizing the weighted mean

$$\begin{aligned} & \int \mathbb{E}_{\mathbf{x}_0} [p(\mathbf{x}_t | \mathbf{x}_0) \|\mathbf{s}_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{x}_0)\|^2] d\mathbf{x}_t \\ &= \mathbb{E}_{\mathbf{x}_0, \mathbf{x}_t \sim p(\mathbf{x}_t | \mathbf{x}_0) \bar{p}(\mathbf{x}_0)} [\|\mathbf{s}_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{x}_0)\|^2] \end{aligned}$$

This is called conditional score matching or denoising score matching.

Here we want to prove that this is equivalent to the original score matching.

3.2 Score Matching is Conditional Score Matching

First, the target of score matching is

$$\mathbb{E}_{x_t \sim p_t(x_t)} [\|\nabla_{x_t} \log p_t(x_t) - s_\theta(x_t, t)\|^2]$$

Previous equation shows

$$p(\mathbf{x}_t) = \int p(\mathbf{x}_t | \mathbf{x}_0) \tilde{p}(\mathbf{x}_0) d\mathbf{x}_0 = \mathbb{E}_{\mathbf{x}_0} [p(\mathbf{x}_t | \mathbf{x}_0)]$$

Here $p_0(x_0)$ is the training data distributon,

$$\nabla_{\mathbf{x}_t} \log(p(\mathbf{x}_t)) = \frac{\mathbb{E}_{\mathbf{x}_0} [\nabla_{\mathbf{x}_t} p(\mathbf{x}_t | \mathbf{x}_0)]}{\mathbb{E}_{\mathbf{x}_0} [p(\mathbf{x}_t | \mathbf{x}_0)]}$$

By our assumption, we know $\nabla_{\mathbf{x}_t} p(\mathbf{x}_t | \mathbf{x}_0)$ and $p(\mathbf{x}_t | \mathbf{x}_0)$ analytically, thus we can approximate $\nabla_{\mathbf{x}_t} \log(p(\mathbf{x}_t))$ by sampling and forward calculation.

Here we give a example:

The data distribution $p_0(\boldsymbol{x})$, the prior distribution $p_T(\boldsymbol{x})$, we choose the following SDE

$$d\boldsymbol{x} = \boldsymbol{\sigma}_t d\boldsymbol{W}_t, t \in [0, 1]$$

In this case

$$p_{(\boldsymbol{x}(t)|\boldsymbol{x}(0))} = \mathcal{N}\left(\boldsymbol{x}(t); \boldsymbol{x}(0), \frac{1}{2\log\sigma}(\sigma^{2t} - 1)I\right)$$

Then the gradient $\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t | \boldsymbol{x}_0)$ is easily calculated.

3.3 Inequality

$$\begin{aligned}\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) &= \frac{\nabla_{\mathbf{x}_t} p_t(\mathbf{x}_t)}{p_t(\mathbf{x}_t)} \\&= \frac{\int p_0(\mathbf{x}_0) \nabla_{\mathbf{x}_t} p_t(\mathbf{x}_t | \mathbf{x}_0) d\mathbf{x}_0}{p_t(\mathbf{x}_t)} \\&= \frac{\int p_0(\mathbf{x}_0) p_t(\mathbf{x}_t | \mathbf{x}_0) \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t | \mathbf{x}_0) d\mathbf{x}_0}{p_t(\mathbf{x}_t)} \\&= \int p_t(\mathbf{x}_0 | \mathbf{x}_t) \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t | \mathbf{x}_0) d\mathbf{x}_0 \\&= \mathbb{E}_{\mathbf{x}_0 \sim p_t(\mathbf{x}_0 | \mathbf{x}_t)} [\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t | \mathbf{x}_0)]\end{aligned}$$

(2)

Conditional Score Matching is a upper bound of Score Matching:

$$\begin{aligned} & \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2] \\ = & \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} [\|\mathbb{E}_{\mathbf{x}_0 \sim p_t(\mathbf{x}_0|\mathbf{x}_t)} [\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{x}_0)] - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2] \\ \leq & \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} \mathbb{E}_{\mathbf{x}_0 \sim p_t(\mathbf{x}_0|\mathbf{x}_t)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{x}_0) - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2] \\ = & \mathbb{E}_{\mathbf{x}_0 \sim p_0(\mathbf{x}_0), \mathbf{x}_t \sim p_t(\mathbf{x}_t|\mathbf{x}_0)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{x}_0) - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2] \end{aligned}$$

(3)

3.4 Equivalence

First, the score matching suggests that

$$\begin{aligned} & \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2] \\ &= \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)\|^2 + \|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2 - 2 \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) \cdot \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)] \end{aligned}$$

(4)

The Conditional Score Matching suggests that

$$\begin{aligned} & \mathbb{E}_{\mathbf{x}_0, \mathbf{x}_t \sim p_0(\mathbf{x}_0) p_t(\mathbf{x}_t | \mathbf{x}_0)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t | \mathbf{x}_0) - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2] \\ &= \mathbb{E}_{\mathbf{x}_0, \mathbf{x}_t \sim p_0(\mathbf{x}_0) p_t(\mathbf{x}_t | \mathbf{x}_0)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t | \mathbf{x}_0)\|^2 + \|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2 - 2 \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) \cdot \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t | \mathbf{x}_0)] \\ &= \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t), \mathbf{x}_0 \sim p_t(\mathbf{x}_0 | \mathbf{x}_t)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t | \mathbf{x}_0)\|^2 + \|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2 - 2 \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) \cdot \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t | \mathbf{x}_0)] \end{aligned}$$

(5)

$$\begin{aligned}
&= \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} [\mathbb{E}_{\mathbf{x}_0 \sim p_t(\mathbf{x}_0|\mathbf{x}_t)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{x}_0)\|^2]] + \|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2 \\
&- 2 \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) \cdot \mathbb{E}_{\mathbf{x}_0 \sim p_t(\mathbf{x}_0|\mathbf{x}_t)} [\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{x}_0)] \\
&= \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} [\mathbb{E}_{\mathbf{x}_0 \sim p_t(\mathbf{x}_0|\mathbf{x}_t)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{x}_0)\|^2] + \|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|^2] \\
&- \mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} [2 \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) \cdot \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)]
\end{aligned}$$

The difference between two equation is

$$\mathbb{E}_{\mathbf{x}_t \sim p_t(\mathbf{x}_t)} [\mathbb{E}_{\mathbf{x}_0 \sim p_t(\mathbf{x}_0|\mathbf{x}_t)} [\|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|\mathbf{x}_0)\|^2] - \|\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)\|^2] \quad (6)$$

Which is independent of training parameters.

4 Recap of Poisson Flow Model

Suppose data sample $\mathbf{x} \in \mathbb{R}^d$, augmented data sample $(\mathbf{x}, t) \in \mathbb{R}^{d+1}$, then the distribution become $\mathbf{x} \sim \tilde{p}(\mathbf{x}) \rightarrow (\mathbf{x}, t) \sim \delta(t)\tilde{p}(\mathbf{x})$. Now the gravity field is calculated as

$$\begin{aligned}\mathbf{F}(\mathbf{x}, t) &= -\frac{1}{S_{d+1}(1)} \int \frac{(\mathbf{x} - \mathbf{x}_0, t - t_0)}{(|\mathbf{x} - \mathbf{x}_0|^2 + (t - t_0)^2)^{(d+1)/2}} \delta(t_0) \tilde{p}(\mathbf{x}_0) d\mathbf{x}_0 dt_0 \\ &= -\frac{1}{S_{d+1}(1)} \int \frac{(\mathbf{x} - \mathbf{x}_0, t)}{(|\mathbf{x} - \mathbf{x}_0|^2 + t^2)^{(d+1)/2}} \tilde{p}(\mathbf{x}_0) d\mathbf{x}_0 \\ &\triangleq (\mathbf{F}_{\mathbf{x}}, \mathbf{F}_t)\end{aligned}$$

where $\mathbf{F}_{\mathbf{x}}$ is the first d elements of $\mathbf{F}(\mathbf{x}, t)$, while \mathbf{F}_t is the last element of $\mathbf{F}(\mathbf{x}, t)$.

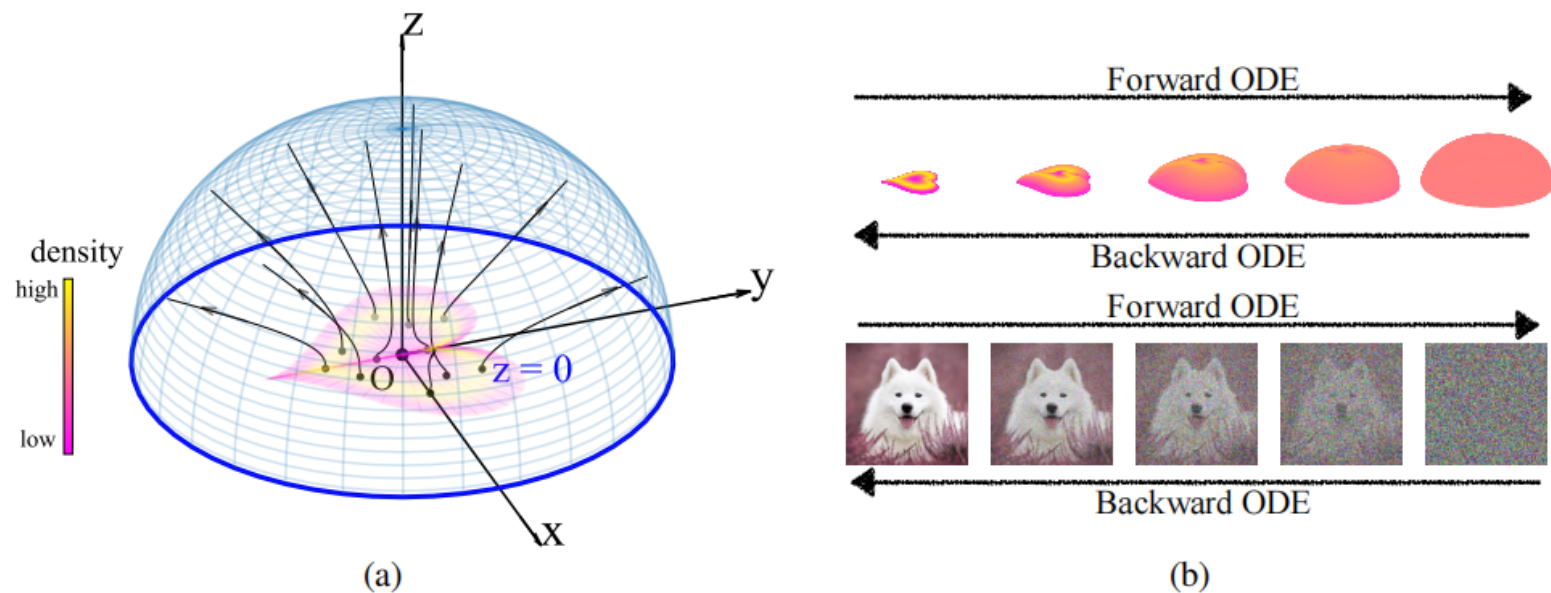


Figure 1: **(a)** 3D Poisson field trajectories for a heart-shaped distribution **(b)** The evolutions of a distribution (**top**) or an (augmented) sample (**bottom**) by the forward/backward ODEs pertained to the Poisson field.

If $F(\mathbf{x}, t)$ is known,

$$(d\mathbf{x}, dt) = (\mathbf{F}_{\mathbf{x}}, \mathbf{F}_t)d\tau \Rightarrow \frac{d\mathbf{x}}{dt} = \frac{\mathbf{F}_{\mathbf{x}}}{\mathbf{F}_t}$$

will generate the desire data by sampling from $t = T$ and compute ODE backward. The prior is not uniform

$$p_{\text{prior}}(\mathbf{x}) \propto \frac{1}{(\|\mathbf{x}\|^2 + T^2)^{(d+1)/2}}$$

The gravity field is

$$\mathbb{E}_{\mathbf{x}_0 \sim \tilde{p}(\mathbf{x}_0)} \left[-\frac{(\mathbf{x} - \mathbf{x}_0, t)}{(\|\mathbf{x} - \mathbf{x}_0\|^2 + t^2)^{\frac{d+1}{2}}} \right]$$

Then we use $\mathbb{E}_{\mathbf{x}}[\mathbf{x}] = \arg \min_{\boldsymbol{\mu}} \mathbb{E}_{\mathbf{x}}[\|\mathbf{x} - \boldsymbol{\mu}\|^2]$ to design loss

$$\mathbb{E}_{\mathbf{x}_0 \sim \tilde{p}(\mathbf{x}_0)} \left[\left| \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}, t) + \frac{(\mathbf{x} - \mathbf{x}_0, t)}{(|\mathbf{x} - \mathbf{x}_0|^2 + t^2)^{(d+1)/2}} \right|^2 \right]$$

4.1 Forward sampling

To see the Poisson Flow model more clearly, we derive the model using Green's function method.

Consider $\mathbf{x}_t \in \mathbb{R}^d$, $t \in [0, T]$, an PF-ODE $\frac{d\mathbf{x}_t}{dt} = \mathbf{f}_t(\mathbf{x}_t)$ convert the probability density with

$$\frac{\partial p_t(\mathbf{x}_t)}{\partial t} + \nabla_{\mathbf{x}_t}(\mathbf{f}_t(\mathbf{x}_t)p_t(\mathbf{x}_t)) = 0$$

which is equivalent to

$$\underbrace{\left(\frac{\partial}{\partial t}, \nabla_{\mathbf{x}_t}\right)}_{\nabla_{(t, \mathbf{x}_t)}} \cdot \underbrace{(p_t(\mathbf{x}_t), \mathbf{f}_t(\mathbf{x}_t)p_t(\mathbf{x}_t))}_{\mathbf{u} \in \mathbb{R}^{d+1}} = 0 \quad (7)$$

let $\mathbf{u}(t, \mathbf{x}_t) = (p_t(\mathbf{x}_t), \mathbf{f}_t(\mathbf{x}_t)p_t(\mathbf{x}_t))$, the equation is then $\nabla_{(t, \mathbf{x}_t)}\mathbf{u}(t, \mathbf{x}_t) = 0$ with $\frac{d\mathbf{x}_t}{dt} = \mathbf{f}_t(\mathbf{x}_t) = \frac{u_{>1}}{u_1}(t, \mathbf{x}_t)$ and boundary condition

$$\left\{ \begin{array}{ll} \mathbf{u}_1(0, \mathbf{x}_0) = p_0(\mathbf{x}_0) & \text{(初值条件)} \\ \int \mathbf{u}_1(t, \mathbf{x}_t) d\mathbf{x}_t = 1 & \text{(积分条件)} \end{array} \right.$$

Using Green's function, we first try to solve

$$\begin{cases} \nabla_{(t, \mathbf{x}_t)} \cdot \mathbf{G}(t, 0; \mathbf{x}_t, \mathbf{x}_0) = 0 \\ \mathbf{G}_1(0, 0; \mathbf{x}_t, \mathbf{x}_0) = \delta(\mathbf{x}_t - \mathbf{x}_0), \int \mathbf{G}_1(t, 0; \mathbf{x}_t, \mathbf{x}_0) d\mathbf{x}_t = 1 \end{cases}$$

If we have such Green's function, then

$$\mathbf{u}(t, \mathbf{x}_t) = \int \mathbf{G}(t, 0; \mathbf{x}_t, \mathbf{x}_0) p_0(\mathbf{x}_0) d\mathbf{x}_0 = \mathbb{E}_{\mathbf{x}_0 \sim p_0(\mathbf{x}_0)}[\mathbf{G}(t, 0; \mathbf{x}_t, \mathbf{x}_0)] \quad (10)$$

Here $G_1(t, 0; \mathbf{x}_t, \mathbf{x}_0)$ is $p_t(\mathbf{x}_t | \mathbf{x}_0)$.

For Poisson Flow model:

$$\mathbf{G}(t, 0; \mathbf{x}_t, \mathbf{x}_0) = C \times \frac{(t, \mathbf{x}_t - \mathbf{x}_0)}{(t^2 + \|\mathbf{x}_t - \mathbf{x}_0\|^2)^{(d+1)/2}} \quad (11)$$

$$p_t(\mathbf{x}_t | \mathbf{x}_0) \propto \frac{t}{(t^2 + \|\mathbf{x}_t - \mathbf{x}_0\|^2)^{(d+1)/2}} \quad (12)$$

Thus as T become large enough, the prior is

$$p_{prior}(\mathbf{x}_T) \propto \frac{T}{(T^2 + \|\mathbf{x}_T\|^2)^{(d+1)/2}} \quad (13)$$

By substitution $\mathbf{z} = \frac{(\mathbf{x}_t - \mathbf{x}_0)}{t}$, the posterior $p_t(\mathbf{x}_t|\mathbf{x}_0)$ is sampled by

$$p(\mathbf{z}) \propto \frac{1}{(1 + \|\mathbf{z}\|^2)^{(d+1)/2}} \quad (14)$$

and $\mathbf{x}_t = \mathbf{x}_0 + t\mathbf{z}$, which is exactly

$$\mathbf{x} = \mathbf{x}_0 + \|\boldsymbol{\varepsilon}_x\|(1 + \tau)^m \mathbf{u}, t = |\varepsilon_t|(1 + \tau)^m$$

where $(\boldsymbol{\varepsilon}_x, \boldsymbol{\varepsilon}_t) \sim \mathcal{N}(0, \sigma^2 I_{(d+1) \times (d+1)})$, $m \sim U[0, M]$, u uniform random variable of unit element of d -sphere.

5 GENPHYS

The author propose a framework that can convert physical PDEs to generative models, termed Generative Models from Physical Processes (GenPhys). For a PDE p , we denote the corresponding generative model p -GenPhys. We will show that p -GenPhys is a generative model if the PDE p is s -generative (s for smooth):

1. p is equivalent to a density flow.
2. The solution of p becomes "smoother" over time.

And the pde canbe categorize into three classes: s -generative, conditionally s -generative (depending on some coefficients in PDE), or not s -generative.

1. p is s -generative. Examples: Diffusion, Poisson, Yukawa (screened Poisson), biharmonic, fractal diffusion, higher-order diffusion.
2. p is conditionally s -generative. Examples: Dissipative wave, Helmholtz.
3. p is not s -generative. Examples: Ideal wave, Schrodinger, Dirac.

5.1 Generative Model from Physical Process

Algorithm 1: Generative models from physical processes (GenPhys)

Input : partial differential equation $\hat{L}\phi(\mathbf{x}, t) = 0$, data distribution $p_{\text{data}}(\mathbf{x})$

Output: generated samples \mathbf{x}

- 1 (1) Rewrite $\hat{L}\phi(\mathbf{x}, t)$ in the form $\frac{\partial p(\mathbf{x}, t)}{\partial t} + \nabla \cdot [p(\mathbf{x}, t)\mathbf{v}(\mathbf{x}, t)] - R(\mathbf{x}, t)$ such that $p = p(\phi, \phi_t, \nabla\phi, \dots)$, $\mathbf{v} = \mathbf{v}(\phi, \phi_t, \nabla\phi, \dots)$, $R = R(\phi, \phi_t, \nabla\phi, \dots)$;
 - 2 (2) Solve $\hat{L}\phi(\mathbf{x}, t) = p_{\text{data}}(\mathbf{x})\delta(t)$. If \hat{L} is linear, we can express $\phi(\mathbf{x}, t)$ in terms of the Green's function $G(\mathbf{x}, t; \mathbf{x}')$: $\phi(\mathbf{x}, t) = \int G(\mathbf{x}, t; \mathbf{x}')p_{\text{data}}(\mathbf{x}')d^N\mathbf{x}'$, where $\hat{L}G(\mathbf{x}, t; \mathbf{x}') = \delta(\mathbf{x} - \mathbf{x}')\delta(t)$;
 - 3 (3) Using the relations in (1) and solutions in (2) to obtain $p(\mathbf{x}, t)$, $\mathbf{v}(\mathbf{x}, t)$, $R(\mathbf{x}, t)$;
 - 4 (4) Train a neural network $\mathbf{s}_\theta(\mathbf{x}, t)$ to fit $\mathbf{v}(\mathbf{x}, t)$ such that $\mathbf{s}_\theta(\mathbf{x}, t) \approx \mathbf{v}(\mathbf{x}, t)$. Train another neural network $W_\alpha(\mathbf{x}, t)$ to fit $R(\mathbf{x}, t)$ such that $W_\alpha(\mathbf{x}, t) \approx R(\mathbf{x}, t)$;
 - 5 (5) Draw $\mathbf{x}(T) \sim p(\mathbf{x}, T)$, simulate $\frac{d\mathbf{x}(t)}{dt} = \mathbf{s}_\theta(\mathbf{x}, t)$ from $t = T$ to $t = 0$ with the branching process $W_\alpha(\mathbf{x}, t)$. Output $\mathbf{x}(0)$.
-

5.1.1 GENERATIVE MODELS AS DENSITY FLOW

Given i.i.d data samples from $p_{\text{data}}(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^N$, the goal is to obtain new sample from $p_{\text{data}}(\mathbf{x})$. A continuous physical process $\frac{d\mathbf{x}}{dt} = \mathbf{v}(\mathbf{x}, t)$ evolves the probability distribution as

$$\frac{\partial p(\mathbf{x}, t)}{\partial t} + \nabla \cdot [p(\mathbf{x}, t)\mathbf{v}(\mathbf{x}, t)] = 0$$

known as probability flow equation or continuity equation. One can draw samples from $p(\mathbf{x}, T)$ and run process backwardly.

We further allow non-conservation term:

$$\frac{\partial p(\mathbf{x}, t)}{\partial t} + \nabla \cdot [p(\mathbf{x}, t)\mathbf{v}(\mathbf{x}, t)] - R(\mathbf{x}, t) = 0$$

In this case the $p(\boldsymbol{x}, t)$ is only a density distribution instead of probability distribution and the equation is density flow equation.

We aim to design $p(\boldsymbol{x}, t), \boldsymbol{v}(\boldsymbol{x}, t), R(\boldsymbol{x}, t)$ such that the initial and final boundary conditions are met: $p(\boldsymbol{x}, 0) = p_{\text{data}}(\boldsymbol{x}), p_{\text{prior}}(\boldsymbol{x}) = p(\boldsymbol{x}, T)$ which is asymptotically independent of $p_{\text{data}}(\boldsymbol{x})$ as $T \rightarrow \infty$.

It's convenient to include the initial condition as a source term

$$\hat{M}(p, \boldsymbol{v}, R) = \frac{\partial p(\boldsymbol{x}, t)}{\partial t} + \nabla \cdot [p(\boldsymbol{x}, t) \boldsymbol{v}(\boldsymbol{x}, t)] - R(\boldsymbol{x}, t) = p_{\text{data}}(\boldsymbol{x}) \delta(t)$$

(Not Clear)

5.1.2 PHYSICAL PROCESSES AND PHYSICAL PDES

Continuous physical processes are described by partial differential equations

$$\hat{L}\phi = F(\phi, \phi_t, \phi_{tt}, \nabla\phi, \nabla^2\phi, \dots) = f(\mathbf{x}, t)$$

$D\phi(\mathbf{x}, t) \subset \mathbb{R}^N \times \mathbb{R}^+$. We only discuss linear PDEs and also symmetric both in space and time, where F does not depend explicitly on \mathbf{x} or t .

For linear PDEs, solution $\phi(\mathbf{x}, t)$ can be expressed as a convolution of the Green's function $G(\mathbf{x}, t; \mathbf{x}', t')$ with the source term $f(\mathbf{x}, t)$:

$$\phi(\mathbf{x}, t) = \int G(\mathbf{x}, t; \mathbf{x}', t') f(\mathbf{x}', t') d^N \mathbf{x}' dt'$$

The Green's function $G(\mathbf{x}, t; \mathbf{x}', t')$ is defined as the solution of

$$\hat{L}G(\mathbf{x}, t; \mathbf{x}', t') = \delta(\mathbf{x} - \mathbf{x}')\delta(t - t'), t > t'; G(\mathbf{x}, t; \mathbf{x}', t') = 0, t < t'$$

5.1.3 Converting Physical PDE to Density Flows

Setting $f(\mathbf{x}, t) = p_{\text{data}}(\mathbf{x})\delta(t)$. By setting $p = p(\phi, \phi_t, \nabla\phi, \dots)$, $\mathbf{v} = \mathbf{v}(\phi, \phi_t, \nabla\phi, \dots)$, $R = R(\phi, \phi_t, \nabla\phi, \dots)$ we should modify physical PDE into Density flows form.

1. Well-behaved density flow: $p(\mathbf{x}, t)$ is a density distribution, i.e., $p(\mathbf{x}, t) \geq 0$. In addition,

(p, \mathbf{v}, R) should be well-behaved (e.g., cannot be discontinuous or have singularities etc.)

2. Smooth PDEs: As $T \rightarrow \infty$, the final distribution $p(\mathbf{x}, T)$ becomes asymptotically independent of the initial distribution $p(\mathbf{x}, 0) = p_{\text{data}}(\mathbf{x})$.

If a PDE satisfies both (C1) and (C2), we call it s -generative, where s stands for smooth.

5.1.4 Diffusion model

We first convert diffusion equation $\phi_t - \nabla^2 \phi = p_{\text{data}}(\mathbf{x})\delta(t)$ to a density flow $\frac{\partial p}{\partial t} + \nabla \cdot (p\mathbf{v}) - R = p_{\text{data}}(\mathbf{x})\delta(t)$:

$$\phi_t - \nabla^2 \phi = \frac{\partial \phi}{\partial t} + \nabla \cdot (\phi(-\nabla \log \phi)) - 0 = 0 \Leftrightarrow \frac{\partial p}{\partial t} + \nabla \cdot (p\mathbf{v}) - R = 0$$

We have:

$$p = \phi, \mathbf{v} = -\nabla \log \phi, R = 0$$

The solution ϕ of diffusion equation is

$$\phi(\mathbf{x}, t) = \int G(\mathbf{x}, t; \mathbf{x}') p_{\text{data}}(\mathbf{x}') d^N x', G(\mathbf{x}, t; \mathbf{x}') = \frac{1}{(2\pi t)^{\frac{N}{2}}} \exp\left(-\frac{|\mathbf{x} - \mathbf{x}'|^2}{2t}\right)$$

$$\begin{aligned}
p(\mathbf{x}, t) &= \phi(\mathbf{x}, t) = \int G(\mathbf{x}, t; \mathbf{x}') p_{\text{data}}(\mathbf{x}') d^N \mathbf{x}' \\
&= \frac{1}{(2\pi t)^{N/2}} \int p_{\text{data}}(\mathbf{x}') \exp\left(-\frac{|\mathbf{x} - \mathbf{x}'|^2}{2t}\right) d^N \mathbf{x}' \\
\mathbf{v}(\mathbf{x}, t) &= -\nabla \log \phi(\mathbf{x}, t) = -\frac{1}{p(\mathbf{x}, t)} \int \nabla G(\mathbf{x}, t; \mathbf{x}') p_{\text{data}}(\mathbf{x}') d^N \mathbf{x}' \\
&= \mathbb{E}_{p_t(\mathbf{x}'|\mathbf{x})} \left(\frac{\mathbf{x} - \mathbf{x}'}{t} \right) \\
R(\mathbf{x}, t) &= 0
\end{aligned}$$

here $p_t(\mathbf{x}'|\mathbf{x}) \propto p_{\text{data}}(\mathbf{x}') G(\mathbf{x}, t; \mathbf{x}') \sim p_{\text{data}}(\mathbf{x}') \exp\left(-\frac{|\mathbf{x} - \mathbf{x}'|^2}{2t}\right)$, note that $-\mathbf{v}(\mathbf{x}, t)$ recover score function $\nabla_{\mathbf{x}} \log p(\mathbf{x}, t)$.

To check (C2), we define F to measure the independency between final distribution and initial condition:

$$F(\mathbf{x}'_1, \mathbf{x}'_2, T) = \int \sqrt{p(\mathbf{x}, T; \mathbf{x}'_1)} \sqrt{p(\mathbf{x}, T; \mathbf{x}'_2)} d^N \mathbf{x} = \exp\left(-\frac{|\mathbf{x}_1 - \mathbf{x}_2|^2}{8T}\right)$$

where $F=1$ means independence, $F=0$ means dependence. We have

$$\lim_{T \rightarrow \infty} F(\boldsymbol{x}'_1, \boldsymbol{x}'_2, T) \rightarrow 1$$

5.1.5 Poisson Flow Generative Models

We aim to convert Poisson equation $-(\phi_{tt} + \nabla^2 \phi) = p_{\text{data}}(\mathbf{x})\delta(t)$ to a density flow $\frac{\partial p}{\partial t} + \nabla \cdot (p\mathbf{v}) - R = p_{\text{data}}(\mathbf{x})\delta(t)$

$$-(\phi_{tt} + \nabla^2 \phi) = \frac{\partial(-\phi_t)}{\partial t} + \nabla \cdot \left[-\phi_t \left(\frac{\nabla \phi}{\phi_t} \right) \right] - 0 = 0 \Leftrightarrow \frac{\partial p}{\partial t} + \nabla \cdot (p\mathbf{v}) - R = 0$$

Comparing the two sides gives:

$$p = \phi_t, \mathbf{v} = \frac{\nabla \phi}{\phi_t}, R = 0$$

Note that we are reinterpreting as time what physicists consider as one of the $N + 1$ spatial dimensions. For $N > 2$

$$\phi(\mathbf{x}, t) = \int G(\mathbf{x}, t; \mathbf{x}') p_{\text{data}}(\mathbf{x}') d^N x', G(\mathbf{x}, t; \mathbf{x}') = \frac{1}{A_N} \frac{1}{(t^2 + |\mathbf{x} - \mathbf{x}'|^2)^{\frac{N-1}{2}}}$$

where A_N is the surface area of a unit N -sphere.

$$p(\mathbf{x}, t) = -\phi_t(\mathbf{x}, t) = \int -G_t(\mathbf{x}, t; \mathbf{x}') p_{\text{data}}(\mathbf{x}') d^N \mathbf{x}'$$

$$= \frac{N-1}{2 A_N} \int \frac{t}{(t^2 + |\mathbf{x} - \mathbf{x}'|^2)^{\frac{N+1}{2}}} p_{\text{data}}(\mathbf{x}') d^N \mathbf{x}'$$

$$\mathbf{v}(\mathbf{x}, t) = \frac{\nabla \phi(\mathbf{x}, t)}{\phi_t(\mathbf{x}, t)} = \frac{1}{p(\mathbf{x}, t)} \int \nabla G(\mathbf{x}, t; \mathbf{x}') p_{\text{data}}(\mathbf{x}') d^N \mathbf{x}' = \mathbb{E}_{p_t(\mathbf{x}'|\mathbf{x})} \left(\frac{\mathbf{x} - \mathbf{x}'}{t} \right)$$

$$R(\mathbf{x}, t) = 0$$

here $p_t(\mathbf{x}'|\mathbf{x}) \propto p_{\text{data}}(\mathbf{x}') G(\mathbf{x}, t; \mathbf{x}') \sim p_{\text{data}}(\mathbf{x}') \frac{1}{(t^2 + |\mathbf{x} - \mathbf{x}'|^2)^{\frac{N-1}{2}}}$, note that $\mathbf{v}(\mathbf{x}, t)$ recover poisson fields.

We check C2 holds. We notice that for large T:

$$p(\mathbf{x}, T; \mathbf{x}') \sim \frac{1}{\left(1 + \frac{|\mathbf{x} - \mathbf{x}'|^2}{T^2}\right)^{\frac{N+1}{2}}} \approx \exp\left(-\frac{(N+1)|\mathbf{x} - \mathbf{x}'|^2}{2T^2}\right)$$

we can show that $\lim_{T \rightarrow \infty} F(\mathbf{x}'_1, \mathbf{x}'_2, T) \rightarrow 1$.

5.2 WHICH PHYSICAL PROCESSES CAN BE GENPHYS

5.2.1 Ideal wave equation (not s -generative)

$\phi_{tt} - \nabla^2 \phi = 0$ describing the propagation of waves of sound, light. Rewritten as $\frac{\partial(-\phi_t)}{\partial t} + \nabla \cdot \left((-\phi_t) \left(-\frac{\nabla \phi}{\phi_t} \right) \right) - 0 = 0$, we have $p = -\phi_t$, $\mathbf{v} = \frac{-\nabla \phi}{\phi_t}$, $R = 0$. The Green function for $N=2$ is $G(\mathbf{x}, t; \mathbf{x}') = \Theta \frac{(t-r)}{\sqrt{t^2 - r^2}}$, where Θ is the step function, $r = |\mathbf{x} - \mathbf{x}'|$. (C1) fails since \mathbf{v} diverges at wave front. (C2) fails since the wave front preserves initial information along the way, so the final distribution is dependent on the initial distribution.

5.2.2 Dissipative wave equation (conditionally s -generative)

$\phi_{tt} + 2\varepsilon\phi_t - \nabla^2\phi = 0$ where ε is the damping coefficient. It describes the wave propagation with dissipation. Rewritten as

$$\frac{\partial(-\phi_t - 2\varepsilon\phi)}{\partial t} + \nabla \cdot \left((-\phi_t - 2\varepsilon\phi) \left(-\frac{\nabla\phi}{\phi_t + 2\varepsilon\phi} \right) \right) - 0 = 0$$

we have

$$p = -\phi_t - 2\varepsilon\phi, \mathbf{v} = -\frac{\nabla\phi}{\phi_t + 2\varepsilon\phi}, R = 0$$

Choosing a large enough ε will make the process quantitatively similar to diffusion, so the two conditions should (approximately) hold.

5.2.3 Helmholtz(conditionally s -generative)

$(\nabla_{\bar{x}}^2 + k_0^2)\phi = 0$ is the single frequency wave, here $\bar{x} = (t, \mathbf{x})$. Rewritten as

$$\frac{\partial(-\phi_t)}{\partial t} + \nabla \cdot \left((-\phi_t) \left(\frac{\nabla \phi}{\phi_t} \right) \right) - k_0^2 \phi = 0$$

then we can match by $p = -\phi_t$, $\mathbf{v} = \frac{\nabla \phi}{\phi_t}$, $R = -k_0^2 \phi$. (C1) is conditionally hold. The Green Function is

$$\text{Re}(G(\mathbf{x}, t; \mathbf{x}')) = \frac{1}{4} \left(\frac{k_0}{2\pi \sqrt{t^2 + r^2}} \right)^{\frac{N-1}{2}} Y_{\frac{N-1}{2}} \left(k_0 \sqrt{t^2 + r^2} \right)$$

Note that G is decreasing function of t for small (t, r) since $-Y_{\frac{N-1}{2}} \left(k_0 \sqrt{t^2 + r^2} \right)$ is decreasing (positive) function of t for $\sqrt{t^2 + r^2} \leq r_c$. C2 conditionally hold as long as $k_0 \leq \frac{r_c}{(\sqrt{t^2 + r^2})_{\max}}$.

5.2.4 Screened Poisson Equation (s – generative)

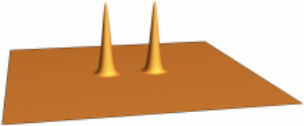
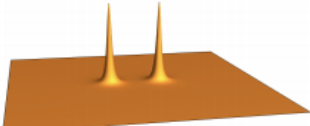
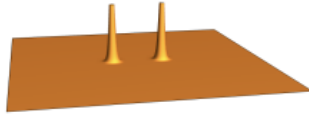
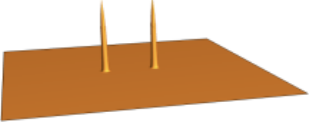
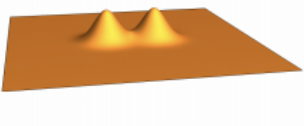
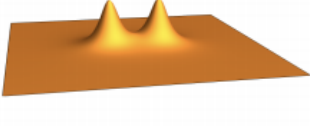
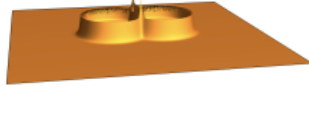
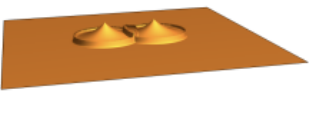
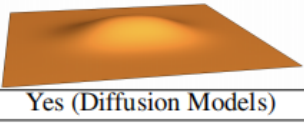

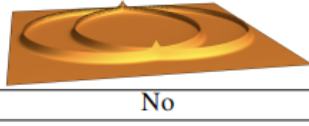

$(\nabla_{\bar{x}}^2 - m^2)\phi = 0$ is the single frequency wave, here $\bar{x} = (t, \mathbf{x})$. Rewritten as

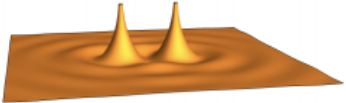
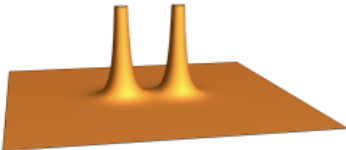
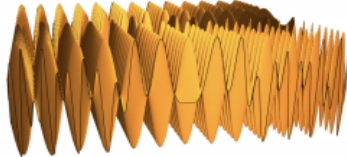
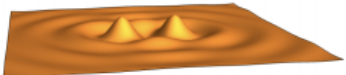
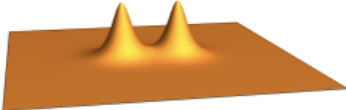
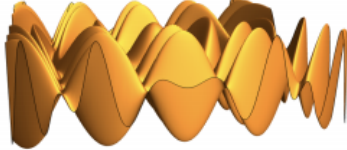



$$\frac{\partial(-\phi_t)}{\partial t} + \nabla \cdot \left((-\phi_t) \left(\frac{\nabla \phi}{\phi_t} \right) \right) + k_0^2 \phi = 0$$

then we can match by $p = -\phi_t$, $\mathbf{v} = \frac{\nabla \phi}{\phi_t}$, $R = m^2 \phi$. The Green function is $G(\mathbf{x}, t; \mathbf{x}') = \left(\frac{m}{\sqrt{t^2 + r^2}} \right)^{\frac{N-1}{2}} K_{\frac{N-1}{2}} \left(m \sqrt{t^2 + r^2} \right)$, K is the modified Bessel function of the second kind.

C1 holds: $p = -\phi_t > 0$ since both $\left(\frac{m}{\sqrt{t^2 + r^2}} \right)^{\frac{N-1}{2}}$ and $K_{\frac{N-1}{2}} \left(m \sqrt{t^2 + r^2} \right)$ is positive and decreasing function of t .

C2 holds since the dispersion relation $w(k) = -i\sqrt{m^2 + k^2}$ satisfies smoothing condition.

equation	diffusion equation	Poisson equation	ideal wave equation	dissipative wave equation
PDE $\hat{L}\phi = 0$	$\phi_t - \nabla^2 \phi = 0$	$\phi_{tt} + \nabla^2 \phi = 0$	$\phi_{tt} - \nabla^2 \phi = 0$	$\phi_{tt} + 2\epsilon\phi_t - \nabla^2 \phi = 0$
rewritten	$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi(-\nabla \log \phi)) = 0$	$\frac{\partial(-\phi_t)}{\partial t} + \nabla \cdot ((-\phi_t)(\frac{\nabla \phi}{\phi_t})) = 0$	$\frac{\partial(-\phi_t)}{\partial t} + \nabla \cdot ((-\phi_t)(-\frac{\nabla \phi}{\phi_t})) = 0$	$\frac{\partial(-\phi_t - 2\epsilon\phi)}{\partial t} + \nabla \cdot ((-\phi_t - 2\epsilon\phi)(\frac{\nabla \phi}{\phi_t + 2\epsilon\phi})) = 0$
p	ϕ	$-\phi_t$	$-\phi_t$	$-(\phi_t + 2\epsilon\phi)$
\mathbf{v}	$-\nabla \log \phi$	$\frac{\nabla \phi}{\phi_t}$	$-\frac{\nabla \phi}{\phi_t}$	$\frac{\nabla \phi}{\phi_t + 2\epsilon\phi}$
R	0	0	0	0
$G(r, t)$	$\frac{1}{(4\pi t)^{\frac{N}{2}}} \exp(-\frac{r^2}{4t})$	$\frac{1}{(t^2 + r^2)^{\frac{N-1}{2}}}$	$\frac{1}{\sqrt{t^2 - r^2}} \Theta(t - r) \text{ (2D)}$	$\frac{e^{-\epsilon t} \cosh(\epsilon \sqrt{t^2 - r^2})}{\sqrt{t^2 - r^2}} \Theta(t - r) \text{ (2D)}$
$\hat{G}(k, t)$	$\exp(-k^2 t)$	$\exp(-kt)$	$\exp(\pm ikt)$	$\exp(-\epsilon t + i\sqrt{k^2 - \epsilon^2} t) \text{ (} k > \epsilon \text{)}$ $\exp(-(\epsilon + \sqrt{k^2 - \epsilon^2})t) \text{ (} k \leq \epsilon \text{)}$
(C1)	Yes	Yes	No	Conditionally yes
(C2)	Yes	Yes	No	Conditionally yes
Illustration ϕ				
				
				
s-generative?	Yes (Diffusion Models)	Yes (Poisson Flow)	No	Conditionally Yes (large ϵ)

equation	Helmholtz equation	screened Poisson equation (Yukawa)	Schrödinger equation
PDE $\hat{L}\phi = 0$	$\phi_{tt} + \nabla^2 \phi + k_0^2 \phi = 0$	$\phi_{tt} + \nabla^2 \phi - m^2 \phi = 0$	$i\phi_t + \nabla^2 \phi = 0$
Rewritten	$\frac{\partial(-\phi_t)}{\partial t} + \nabla \cdot ((-\phi_t)(\frac{\nabla \phi}{\phi_t})) - k_0^2 \phi = 0$	$\frac{\partial(-\phi_t)}{\partial t} + \nabla \cdot ((-\phi_t)(\frac{\nabla \phi}{\phi_t})) + m^2 \phi = 0$	$\frac{\partial \phi ^2}{\partial t} + \nabla \cdot (\phi ^2(2\text{Im}\nabla\log\phi)) = 0$
p	$-\phi_t$	$-\phi_t$	$ \phi ^2$
\mathbf{v}	$\frac{\nabla \phi}{\phi_t}$	$\frac{\nabla \phi}{\phi_t}$	$2\text{Im}\nabla\log\phi$
R	$k_0^2 \phi$	$-m^2 \phi$	0
$G(r, t)$	$(\frac{k_0}{\sqrt{t^2 + r^2}})^{\frac{N-1}{2}} H_{\frac{N-1}{2}}^{(1)}(k_0 \sqrt{t^2 + r^2})$	$(\frac{m}{\sqrt{t^2 + r^2}})^{\frac{N-1}{2}} K_{\frac{N-1}{2}}(m \sqrt{t^2 + r^2})$	$\frac{1}{(4\pi i t)^{\frac{N}{2}}} \exp(\frac{ir^2}{4t})$
$\hat{G}(k, t)$	$\exp(-i\sqrt{k_0^2 - k^2}t) \ (k \leq k_0)$ $\exp(-\sqrt{k^2 - k_0^2}t) \ (k > k_0)$	$\exp(-\sqrt{k^2 + m^2}t)$	$\exp(ik^2 t)$
(C1)	Conditional yes	Yes	No
(C2)	Conditional Yes	Yes	No
Illustration ϕ or $\text{Re}\phi$			
			
			
s-generative?	Conditionally yes (small k)	Yes	No

5.3 Dispersion Relation as A Criterion

The dispersion relation relates the wavenumber k and frequency ω . All linear PDEs have wave solutions

$$\phi(\mathbf{x}, t) \propto \exp(-i\omega t) \exp(i\mathbf{k} \cdot \mathbf{x})$$

Physics	PDE	Dispersion Relation	s-generative?
Diffusion	$\phi_t - \nabla^2 \phi = 0$	$\omega = -ik^2$	Yes
Poisson	$\phi_{tt} + \nabla^2 \phi = 0$	$\omega = \pm ik$	Yes
Ideal Wave	$\phi_{tt} - \nabla^2 \phi = 0$	$\omega = \pm k$	No
Dissipative wave	$\phi_{tt} + 2\epsilon\phi_t - \nabla^2 \phi = 0$	$\omega = \begin{cases} i(-\epsilon \pm \sqrt{\epsilon^2 - k^2}) & k \leq \epsilon \\ i\epsilon \pm \sqrt{k^2 - \epsilon^2} & k > \epsilon \end{cases}$	Conditionally Yes (large ϵ)
Helmholtz	$\phi_{tt} + \nabla^2 \phi + k_0^2 \phi = 0$	$\omega = \begin{cases} \pm \sqrt{k_0^2 - k^2} & k \leq k_0 \\ \pm i \sqrt{k^2 - k_0^2} & k > k_0 \end{cases}$	Conditionally Yes (small k_0^2)
Screened Poisson	$\phi_{tt} + \nabla^2 \phi - m^2 \phi = 0$	$\omega = \pm i \sqrt{k^2 + m^2}$	Yes
Schrödinger	$i\phi_t + \nabla^2 \phi = 0$	$\omega = k^2$	No

There is a perfect relation between dispersion relation and s -generative:

1. s -generative: pure imaginary w for all k .
2. conditionally s -generative: pure imaginary w for some large k .
3. not s -generative: have k range with pure imaginary w

定理 9. C2 is equivalent to

$$\text{Im } \omega(k) < \text{Im } \omega(0) \text{ for all } k > 0$$

Define the overlap between $p(\mathbf{x}, t; \mathbf{x}'_i)$ ($i=1,2$)

$$F(\mathbf{x}'_1, \mathbf{x}'_2, t) \equiv \frac{\int p(\mathbf{x}, t; \mathbf{x}'_1) p(\mathbf{x}, t; \mathbf{x}'_2) d^N \mathbf{x}}{\int p(\mathbf{x}, t; \mathbf{x}'_1) p(\mathbf{x}, t; \mathbf{x}'_1) d^N \mathbf{x}},$$

It's an alternative version of previous $F(\mathbf{x}'_1, \mathbf{x}'_2, T) = \frac{\int p(\mathbf{x}, T; \mathbf{x}'_1) p(\mathbf{x}, T; \mathbf{x}'_2) d^N \mathbf{x}}{\int p(\mathbf{x}, T; \mathbf{x}'_1) p(\mathbf{x}, T; \mathbf{x}'_1) d^N \mathbf{x}} \rightarrow 1$, thus C2 requires $\lim_{t \rightarrow \infty} F(\mathbf{x}'_1, \mathbf{x}'_2, t) = 1$.

Define

$$\tilde{p}(\mathbf{k}, t; \mathbf{x}'_i) = \int p(\mathbf{x}, t; \mathbf{x}'_i) \exp(-i\mathbf{k} \cdot \mathbf{x}) d^N \mathbf{x}, \quad i = 1, 2$$

Then $p(\mathbf{x}, t; \mathbf{x}'_i)$ differ only by a translation, so their Fourier function differs only by a phase

$$p(\mathbf{k}, t; \mathbf{x}'_2) = p(\mathbf{k}, t; \mathbf{x}'_1) \exp(i\mathbf{k} \cdot (\mathbf{x}'_1 - \mathbf{x}'_2))$$

$$\begin{aligned} F(\mathbf{x}'_1, \mathbf{x}'_2, t) &= \frac{\int p^*(\mathbf{k}, t; \mathbf{x}'_1) p(\mathbf{k}, t; \mathbf{x}'_2) d^N \mathbf{k}}{\int p^*(\mathbf{k}, t; \mathbf{x}'_1) p(\mathbf{k}, t; \mathbf{x}'_1) d^N \mathbf{k}} \\ &= \frac{\int p^*(\mathbf{k}, t; \mathbf{x}'_1) p(\mathbf{k}, t; \mathbf{x}'_1) \exp(i\mathbf{k} \cdot (\mathbf{x}'_1 - \mathbf{x}'_2)) d^N \mathbf{k}}{\int p^*(\mathbf{k}, t; \mathbf{x}'_1) p(\mathbf{k}, t; \mathbf{x}'_1) d^N \mathbf{k}} \\ &= \frac{\int |p(\mathbf{k}, t; \mathbf{x}'_1)|^2 \cos(\mathbf{k} \cdot (\mathbf{x}'_1 - \mathbf{x}'_2)) d^N \mathbf{k}}{\int |p(\mathbf{k}, t; \mathbf{x}'_1)|^2 d^N \mathbf{k}} \\ &= \langle \cos(\mathbf{k} \cdot (\mathbf{x}'_1 - \mathbf{x}'_2)) \rangle_{|p(\mathbf{k}, t; \mathbf{x}'_1)|^2} \end{aligned}$$

Note that

$$p(\mathbf{k}, t; \mathbf{x}'_1) = \exp(-i\omega(k)t) p(\mathbf{k}, 0; \mathbf{x}'_1) = \exp(-i\text{Re}\omega(k)t) \exp(\text{Im}\omega(k)t) p(\mathbf{k}, t; \mathbf{x}'_1)$$

We have

$$|p(\mathbf{k}, t; \mathbf{x}'_1)|^2 = \exp(2\text{Im}\omega(k)t)$$

Define

$$k^* = \text{argmax}_k \text{Im}(\omega(k))$$

Then since as t increase, $|p(\mathbf{k}, t; \mathbf{x}'_1)|^2$ has increasingly more probability concentrated around $k = k^*$. As a result

$$\lim_{t \rightarrow \infty} F(\mathbf{x}'_1, \mathbf{x}'_2, t) = \langle \cos(\mathbf{k} \cdot (\mathbf{x}'_1 - \mathbf{x}'_2)) \rangle_{|k|=k^*}$$

The limit is 1 iff $k^* = 0$ for $\mathbf{x}'_1 \neq \mathbf{x}'_2$.

The physical interpretation is that waves of $k > 0$ should decay faster than $k = 0$. Note that although dispersion relations may have multiple branches, we only require one branch to satisfy the relation.

6 Consistency Model

6.1 Recap of Score Based Diffusion Model

Diffusion models start by diffusion the $p_{\text{data}}(\mathbf{x})$ with a SDE

$$d\mathbf{x}_t = \boldsymbol{\mu}(\mathbf{x}_t, t)dt + \sigma(t)d\mathbf{w}_t,$$

where $t \in [0, T]$. We denote the distribution of \mathbf{x}_t as $p_t(\mathbf{x}_t)$ and $p_0(\mathbf{x}) = p_{\text{data}}(\mathbf{x})$.

The Probability Flow ODE shows that

$$d\mathbf{x}_t = \left[\boldsymbol{\mu}(\mathbf{x}_t, t) - \frac{1}{2}\sigma(t)^2 \nabla \log p_t(\mathbf{x}_t) \right] dt$$

has the same trajectory distribution with the original SDE.

Typically, one design the SDE such that p_T is close to a tractable Gaussian $\pi(\mathbf{x})$. Here we choose $\boldsymbol{\mu}(\mathbf{x}) = 0$, $\sigma(t) = \sqrt{2t}$, then $p_t(\mathbf{x}) = p_{\text{data}}(\mathbf{x}) \otimes \mathcal{N}(0, t^2 I)$ since

$$p_t(\mathbf{x}(t) | \mathbf{x}(0)) = \mathcal{N}(\mathbf{x}(t); \mathbf{x}(0), [\sigma^2(t) - \sigma^2(0)]I)$$

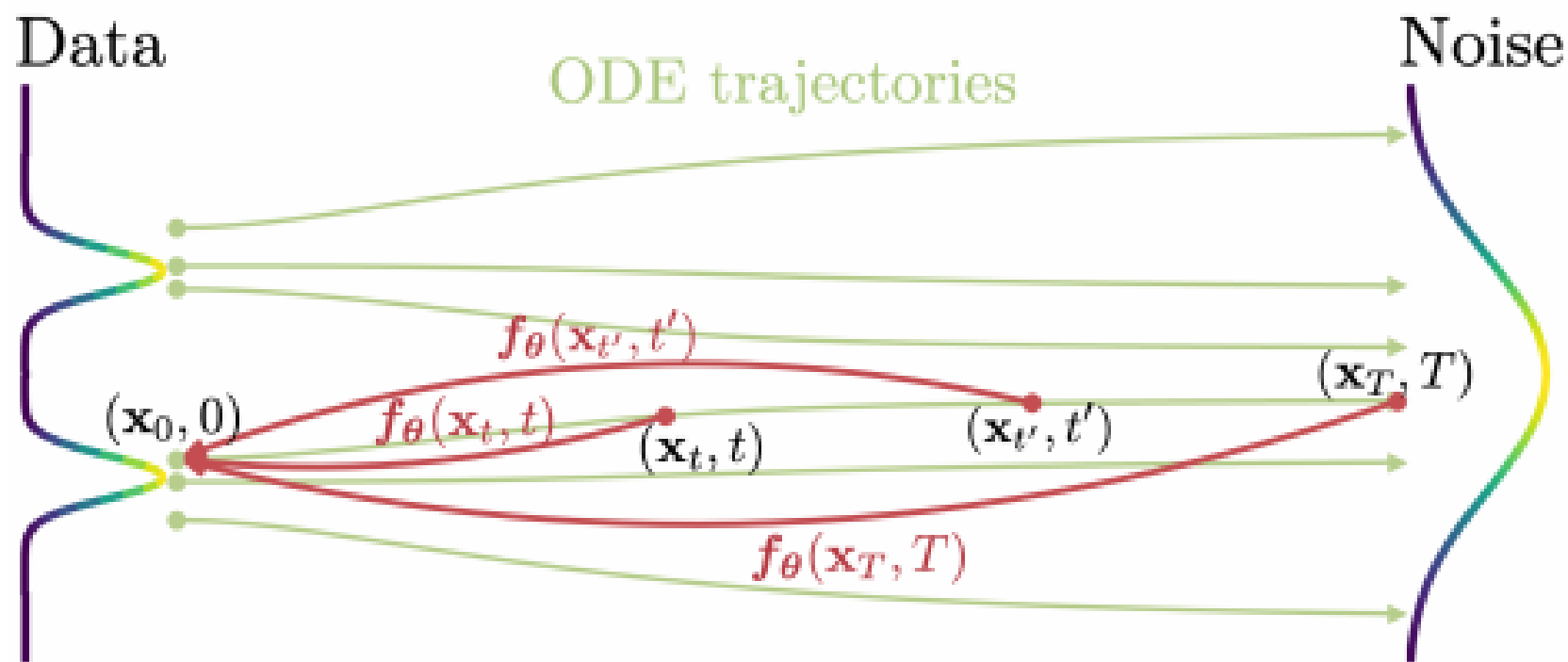


Figure 2: **Consistency models** are trained to map points on any trajectory of the **PF ODE** to the trajectory's origin.

We first train a score model $s_\phi(\mathbf{x}, t) \approx \nabla \log p_t(\mathbf{x})$ via conditional score matching, then sample via

$$\frac{d\mathbf{x}_t}{dt} = -\text{ts}_\phi(\mathbf{x}_t, t)$$

by solving such ODE backwardly. To avoid numerical instability, one typically stops the solver at

$t = \varepsilon$, we rescale the pixel values in images to $[-1, 1]$ and $T = 80, \varepsilon = 0.002$.

6.2 Consistency Models

定义 10. Given a solution trajectory $\{x_t\}_{t \in [0, T]}$ of the PF-ODE , we define the consistency function as $\mathbf{f}: (x_t, t) \rightarrow x_\varepsilon$. A consistency function has the property of self consistency: the outputs are consistent for arbitrary pairs of (x_t, t) that belong to the same PF-ODE, i.e. $\mathbf{f}(x_t, t) = \mathbf{f}(x_{t'}, t')$ for all $t, t' \in [0, T]$. Thus $\mathbf{f}(\cdot, t)$ is a invertible function.

Since $\mathbf{f}(x_\varepsilon, \varepsilon) = x_\varepsilon$, $\mathbf{f}(\cdot, \varepsilon)$ is an identity function. We call this constraint the boundary condition. We parametrize the consistency model using skip connections

$$\mathbf{f}_\theta(\mathbf{x}, t) = c_{\text{skip}}(t)\mathbf{x} + c_{\text{out}}(t)\mathbf{F}_\theta(\mathbf{x}, t)$$

where $c_{\text{skip}}(\varepsilon) = 1$ and $c_{\text{out}}(\varepsilon) = 0$ and both are differentiable.

6.2.1 Sampling

With a well-trained consistency model $f_\theta(\cdot, \cdot)$ we generate samples from $\hat{\mathbf{x}}_T \sim \mathcal{N}(0, T^2 I)$ then evaluating for $\hat{\mathbf{x}}_\epsilon = f_\theta(\hat{\mathbf{x}}_T, T)$.

Also one can choose to evaluate the consistency model multiple times by alternating denoising and noise injection steps for improved sample quality.

Algorithm 1 Multistep Consistency Sampling

Input: Consistency model $f_\theta(\cdot, \cdot)$, sequence of time points $\tau_1 > \tau_2 > \dots > \tau_{N-1}$, initial noise $\hat{\mathbf{x}}_T$

$\mathbf{x} \leftarrow f_\theta(\hat{\mathbf{x}}_T, T)$

for $n = 1$ **to** $N - 1$ **do**

 Sample $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, I)$

$\hat{\mathbf{x}}_{\tau_n} \leftarrow \mathbf{x} + \sqrt{\tau_n^2 - \epsilon^2} \mathbf{z}$

$\mathbf{x} \leftarrow f_\theta(\hat{\mathbf{x}}_{\tau_n}, \tau_n)$

end for

Output: \mathbf{x}

6.3 Training Consistency Models via Distillation

With a pre-trained score model $s_\phi(\mathbf{x}, t)$, we discretizing the time horizon $[\varepsilon, T]$ into $N-1$ sub-intervals: $\{\varepsilon = t_1 < \dots < t_N = T\}$, in practice we follow the formula $t_i = \left(\varepsilon^{\frac{1}{\rho}} + \frac{i-1}{N-1} \left(T^{\frac{1}{\rho}} - \varepsilon^{\frac{1}{\rho}} \right) \right)^\rho$, where $\rho=7$. A numerical ODE solver step is defined by

$$\hat{\mathbf{x}}_{t_n}^\phi = \mathbf{x}_{t_{n+1}} + (t_n - t_{n+1})\Phi(\mathbf{x}_{t_{n+1}}, t_{n+1}; \phi)$$

where Φ represents the update function of ODE solver. For example Euler solver is

$$\hat{\mathbf{x}}_{t_n}^\phi = \mathbf{x}_{t_{n+1}} - (t_n - t_{n+1})s_\phi(\mathbf{x}_{t_{n+1}}, t_{n+1}).$$

One can sample along the distribution, the pair $(\hat{\mathbf{x}}_{t_n}^\phi, \mathbf{x}_{t_{n+1}})$, by first sample \mathbf{x} from dataset, followed by $\mathbf{x}_{t_{n+1}}$ from transition density of SDE $\mathcal{N}(\mathbf{x}, t_{n+1}^2 I)$ and then computing $\hat{\mathbf{x}}_{t_n}^\phi$ by ODE step.

定义 11. The consistency distillation loss is defined as

$$\mathcal{L}_{CD}^N(\boldsymbol{\theta}, \boldsymbol{\theta}^-; \phi) := \mathbb{E} [\lambda(t_n) d(\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{f}_{\boldsymbol{\theta}^-}(\hat{\mathbf{x}}_{t_n}^\phi, t_n))]$$

the expectation is taken wrt $x \sim p_{\text{data}}, n \sim U[1, N-1], \mathbf{x}_{t_{n+1}} \sim \mathcal{N}(\mathbf{x}; t_{n+1}^2 I)$. We refer to $\mathbf{f}_{\boldsymbol{\theta}^-}$ as target network and $\mathbf{f}_{\boldsymbol{\theta}}$ as online network.

定理 12. Let $\Delta t := \max_{n \in [1, N-1]} \{|t_{n+1} - t_n|\}$, and $\mathbf{f}(\cdot, \cdot, \phi)$ be the consistency function of empirical PF-ODE. Assume $\mathbf{f}_{\boldsymbol{\theta}}$ satisfies the Lipschitz condition in L^2 . Assume further that for $n \in [1, N-1]$ the ODE solver called a t_{n+1} has local error uniformly bounded by $O(t_{n+1} - t_n)^{p+1}$, then if $\mathcal{L}_{CD}^N(\boldsymbol{\theta}, \boldsymbol{\theta}; \phi) = 0$ we have

$$\sup_{n, \mathbf{x}} |\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}, t_n) - \mathbf{f}(\mathbf{x}, t_n; \phi)|_2 = O((\Delta t)^p)$$

Algorithm 2 Consistency Distillation (CD)

Input: dataset \mathcal{D} , initial model parameter θ , learning rate η , ODE solver $\Phi(\cdot, \cdot; \phi)$, $d(\cdot, \cdot)$, $\lambda(\cdot)$, and μ

$\theta^- \leftarrow \theta$

repeat

 Sample $\mathbf{x} \sim \mathcal{D}$ and $n \sim \mathcal{U}[1, N - 1]$

 Sample $\mathbf{x}_{t_{n+1}} \sim \mathcal{N}(\mathbf{x}; t_{n+1}^2 \mathbf{I})$

$\hat{\mathbf{x}}_{t_n}^\phi \leftarrow \mathbf{x}_{t_{n+1}} + (t_n - t_{n+1})\Phi(\mathbf{x}_{t_{n+1}}, t_{n+1}; \phi)$

$\mathcal{L}(\theta, \theta^-; \phi) \leftarrow$

$\lambda(t_n)d(f_\theta(\mathbf{x}_{t_{n+1}}, t_{n+1}), f_{\theta^-}(\hat{\mathbf{x}}_{t_n}^\phi, t_n))$

$\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}(\theta, \theta^-; \phi)$

$\theta^- \leftarrow \text{stopgrad}(\mu \theta^- + (1 - \mu)\theta)$

until convergence

6.4 Training Consistency Models in Isolation

引理 13. Let $\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})$, $\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{x}; t^2 I)$ and $p_t(\boldsymbol{x}_t) = p_{\text{data}}(\boldsymbol{x}) \otimes \mathcal{N}(0, t^2 I)$, we have

$$\nabla \log p_t(\boldsymbol{x}) = -\mathbb{E}\left[\frac{\boldsymbol{x}_t - \boldsymbol{x}}{t^2} \middle| \boldsymbol{x}_t\right]$$

证明. $\nabla \log p_t(\mathbf{x}_t) = \nabla_{\mathbf{x}_t} \log \int p_{\text{data}}(\mathbf{x}) p(\mathbf{x}_t | \mathbf{x}) d\mathbf{x}$, where $p(\mathbf{x}_t | \mathbf{x}) = \mathcal{N}(\mathbf{x}_t; \mathbf{x}, t^2 I)$

$$\begin{aligned}
 \nabla \log p_t(\mathbf{x}_t) &= \frac{\int p_{\text{data}}(\mathbf{x}) \nabla_{\mathbf{x}_t} p(\mathbf{x}_t | \mathbf{x}) d\mathbf{x}}{\int p_{\text{data}}(\mathbf{x}) p(\mathbf{x}_t | \mathbf{x}) d\mathbf{x}} \\
 &= \frac{\int p_{\text{data}}(\mathbf{x}) p(\mathbf{x}_t | \mathbf{x}) \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{x}) d\mathbf{x}}{\int p_{\text{data}}(\mathbf{x}) p(\mathbf{x}_t | \mathbf{x}) d\mathbf{x}} \\
 &= \frac{\int p_{\text{data}}(\mathbf{x}) p(\mathbf{x}_t | \mathbf{x}) \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{x}) d\mathbf{x}}{p_t(\mathbf{x}_t)} \\
 &= \int \frac{p_{\text{data}}(\mathbf{x}) p(\mathbf{x}_t | \mathbf{x})}{p_t(\mathbf{x}_t)} \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{x}) d\mathbf{x} \\
 &\stackrel{(i)}{=} \int p(\mathbf{x} | \mathbf{x}_t) \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{x}) d\mathbf{x} \\
 &= \mathbb{E} [\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{x}) | \mathbf{x}_t] \\
 &= -\mathbb{E} \left[\frac{\mathbf{x}_t - \mathbf{x}}{t^2} | \mathbf{x}_t \right]
 \end{aligned}$$

□

定理 14. Let $\Delta t := \max_{n \in [1, N-1]} \{|t_{n+1} - t_n|\}$. Assume d and \mathbf{f}_{θ^-} are both twice differentiable with bounded second derivatives, the weighting function λ is bounded and $\mathbb{E}[\|\nabla \log p_{t_n}(\mathbf{x}_{t_n})\|_2^2] < \infty$. Assume we use Euler solver, and the pre-trained model matches the ground truth ,

$$\forall t \in [\varepsilon, T]: s_\phi(\mathbf{x}, t) = \nabla \log p_t(\mathbf{x})$$

Then

$$L_{\text{CD}}^N(\theta, \theta^-; \phi) = L_{\text{CT}}^N(\theta, \theta^-; \phi) + o(\Delta t)$$

where the expectation is taken wrt $x \sim p_{\text{data}}$, $n \sim U[1, N-1]$, $\mathbf{x}_{t_{n+1}} \sim \mathcal{N}(\mathbf{x}; t_{n+1}^2 I)$. The consistency training objective denoted by $L_{\text{CT}}^N(\theta, \theta^-; \phi)$ is

$$\mathbb{E}[\lambda(t_n) d(\mathbf{f}_\theta(x + t_{n+1} \mathbf{z}, t_{n+1}), \mathbf{f}_{\theta^-}(\mathbf{x} + t_n \mathbf{z}, t_n))]$$

where $\mathbf{z} \sim \mathcal{N}(0, I)$.

Here we can compare with the objective of VE-SDE:

$$\mathbb{E}_{\mathbf{x}_0, \mathbf{x}_t \sim p(\mathbf{x}_t | \mathbf{x}_0) \bar{p}(\mathbf{x}_0)} [|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{x}_0)|^2]$$

$$p(\mathbf{x}(t) | \mathbf{x}(0)) = \mathcal{N}\left(\mathbf{x}(t); \mathbf{x}(0), \frac{1}{2\log\sigma}(\sigma^{2t} - 1)I\right)$$

Algorithm 3 Consistency Training (CT)

Input: dataset \mathcal{D} , initial model parameter $\boldsymbol{\theta}$, learning rate η , step schedule $N(\cdot)$, EMA decay rate schedule $\mu(\cdot)$, $d(\cdot, \cdot)$, and $\lambda(\cdot)$

$\boldsymbol{\theta}^- \leftarrow \boldsymbol{\theta}$ and $k \leftarrow 0$

repeat

 Sample $\mathbf{x} \sim \mathcal{D}$, and $n \sim \mathcal{U}[1, N(k) - 1]$

 Sample $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, I)$

$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-) \leftarrow$

$\lambda(t_n) d(\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x} + t_{n+1}\mathbf{z}, t_{n+1}), \mathbf{f}_{\boldsymbol{\theta}^-}(\mathbf{x} + t_n\mathbf{z}, t_n))$

$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-)$

$\boldsymbol{\theta}^- \leftarrow \text{stopgrad}(\mu(k)\boldsymbol{\theta}^- + (1 - \mu(k))\boldsymbol{\theta})$

$k \leftarrow k + 1$

until convergence

Thank You

