OPTIMAL TRANSPORT-GUIDED CONDITIONAL SCORE-BASED DIFFUSION MODEL

Guangyu Hou

Contents

1 Optimal Transport-Guided Conditional Score-Based Diffusion Model

1.1 Abstract

In this paper, they propose a novel Optimal Transport-guided Conditional Score-based diffusion model (**OTCS**) in this paper. After setting up the coupling relationship for the unpaired or partially paired dataset based on optimal transport, they train the conditional score-based model by designing a "**resampling-by-compatibility**" strategy.
Note:

- The origin method requires the paired data as condition, while here they can tackle the applications with partially paired or even unpaired dataset by optimal transport.
- The **RSC** strategy can choose the sampled data with high compatibility.
- An approach to realize large-scale optimal transport based on diffusion model.
- Applications in unpaired super-resolution and semi-paired image-to-image translation.

1.2 Background

Conditional Scored-based diffusion models (Conditional SBDMs)
This traditional method works for **paired data**, aiming to generate a target sample y from the distribution q of target training data given a condition data x under the classifier guidance or **classifier-free guidance**.

- **Forward Stochastic Differential Equation (SDE):**

$$\mathrm{d}\mathbf{y}_{\mathrm{t}} = f\left(\mathbf{y}_{\mathrm{t}}, t\right)\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}$$

- **Denoising score-matching Loss:**

$$\mathcal{J}_{\mathrm{DSM}}(\theta) = \mathbb{E}_t w_t \mathbb{E}_{\mathbf{y}_0\sim q}\mathbb{E}_{\mathbf{y}_t\sim p_{t|0}(\mathbf{y}_t|\mathbf{y}_0)}\left\| s_\theta\left(\mathbf{y}_t; \mathbf{x}_{\mathrm{cond}}\left(\mathbf{y}_0\right), t\right) - \nabla_{\mathbf{y}_t}\log p_{t|0}\left(\mathbf{y}_t \mid \mathbf{y}_0\right)\right\|_2^2$$

- **Reverse SDE:**

$$\mathrm{d}\mathbf{y}_{\mathrm{t}} = \left[f\left(\mathbf{y}_{\mathrm{t}}, t\right) - g^2(t)s_\theta(\mathbf{y}_{\mathrm{t}}; \mathbf{x}, t)\right]\mathrm{d}t + g(t)\mathrm{d}\overline{\mathbf{w}}$$

Actually, we use a forward stochastic differential equation (SDE) to add Gaussian noises to the target training data for training the model.

$$\mathrm{d}\mathbf{y}_{\mathrm{t}} = f\left(\mathbf{y}_{\mathrm{t}}, t\right)\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}$$

with $\mathbf{y}_0 \sim q$, and $t \in [0, T]$, where $\mathbf{w} \in \mathbb{R}^D$ is a standard Wiener process, $f(\cdot, t) : \mathbb{R}^D \to \mathbb{R}^D$ is the drift coefficient, and $g(t) \in \mathbb{R}$ is the diffusion coefficient.Let $p_{t|0}$ be the conditional distribution of $\mathbf{y}_t$ given the initial state $\mathbf{y}_0$, and p_t be the marginal distribution of $\mathbf{y}_t$. We can choose the $f(\mathbf{y}, t), g(t)$, and T such that $\mathbf{y}_t$ approaches some analytically tractable prior

distribution $p_{\text{prior}}\ (\mathbf{y}_T)$ at time $t = T$, i.e., $p_T\left(\mathbf{y}_T\right) \approx p_{\text{prior}}\ (\mathbf{y}_T)$.

w_t **is the weight for time** t. In this paper, t is uniformly sampled from $[0, T]$. With the trained $s_{\hat{\theta}}(\mathbf{y}; \mathbf{x}, t)$, given a condition data $\mathbf{x}$, the target sample $\mathbf{y}_0$ is generated by the reverse SDE as $\mathrm{d}\mathbf{y}_t = \left[f\left(\mathbf{y}_t, t\right) - g(t)^2 s_{\hat{\theta}}\left(\mathbf{y}_t; \mathbf{x}, t\right)\right] \mathrm{d}t + g(t)\mathrm{d}\overline{\mathbf{w}}$, where $\overline{\mathbf{w}}$ is a standard Wiener process in the reverse-time direction. This process starts from a noise sample $\mathbf{y}_T$ and ends at $t = 0$.

However, in practical applications, we need to face insufficient paired data. Then there exists **two main challenges**: Lacking coupling relationship of data + Unclear formulation for SBDM

1.3 Build Coupling Relationship

Semi-supervised OT.

In semi-supervised OT, a few matched pairs of source and target data points (called "keypoints") $\mathcal{K} = \{(\mathbf{x}_k, \mathbf{y}_k)\}_{k=1}^K$ are given, where K is the number of keypoint pairs. The semi-supervised OT aims to **leverage the given matched keypoints to guide** the correct transport in OT by **preserving the relation of each data point to the keypoints**. Mathematically, we have

$$\min_{\tilde{\pi} \in \tilde{\Gamma}} \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim m \otimes \pi} g(\mathbf{x}, \mathbf{y}), \text{ s.t. } \tilde{\Gamma} = \left\{\tilde{\pi} : T^{\mathbf{x}}_{\#}(m \otimes \tilde{\pi}) = p, T^{\mathbf{y}}_{\#}(m \otimes \tilde{\pi}) = q\right\},$$

where the transport plan $m \otimes \tilde{\pi}$ is $(m \otimes \tilde{\pi})(\mathbf{x}, \mathbf{y}) = m(\mathbf{x}, \mathbf{y})\tilde{\pi}(\mathbf{x}, \mathbf{y})$, and m is a binary mask function. The mask-based modeling of the transport plan ensures that the keypoint pairs are always matched in the derived transport plan (totally paired and un-paired is 1). g, the guiding function, is defined as $g(\mathbf{x}, \mathbf{y}) = d\left(R^s_{\mathbf{x}}, R^t_{\mathbf{y}}\right)$, where $R^s_{\mathbf{x}}, R^s_{\mathbf{y}} \in (0, 1)^K$ model the vector of relation of x, y to each of the paired keypoints in source and target domain respectively, and d is the Jensen-Shannon divergence. The k-th elements of R^s_{x} and R^s_{x} are respectively defined by

$$R^s_{\mathbf{x},k} = \frac{\exp\left(-c\left(\mathbf{x}, \mathbf{x}_k\right)/\tau\right)}{\sum_{l=1}^K \exp\left(-c\left(\mathbf{x}, \mathbf{x}_l\right)/\tau\right)}, R^t_{\mathbf{y},k} = \frac{\exp\left(-c\left(\mathbf{y}, \mathbf{y}_k\right)/\tau\right)}{\sum_{l=1}^K \exp\left(-c\left(\mathbf{y}, \mathbf{y}_l\right)/\tau\right)},$$

where τ is set to 0.1 .

Note that, to ensure feasible solutions, the mass of paired keypoints should be equal, i.e., $p\left(\mathbf{x}_k\right) = q\left(\mathbf{y}_k\right), \forall\left(\mathbf{x}_k, \mathbf{y}_k\right) \in \mathcal{K}$.

1.3.1 L_2-regularized unsupervised and semi-supervised OTs:

For more efficient computation, we present the **L_2-regularized unsupervised and semi-supervised OTs**. The L_2-regularized unsupervised and semi-supervised OTs are respectively given by

$$\min_{\pi \in \Gamma} \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim \pi} c(\mathbf{x}, \mathbf{y}) + \epsilon \chi^2(\pi \| p \times q) \text{ and } \min_{\tilde{\pi} \in \tilde{\Gamma}} \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim m \otimes \pi} g(\mathbf{x}, \mathbf{y}) + \epsilon \chi^2(m \otimes \tilde{\pi} \| p \times q),$$

where $\chi^2(\pi \| p \times q) = \int \frac{\pi(\mathbf{x},\mathbf{y})^2}{p(\mathbf{x})q(\mathbf{y})} \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{y}$, ϵ is regularization factor.

1.3.2 Unified duality:

$$\max_{u,v} \mathcal{F}_{\mathrm{OT}}(u,v) = \mathbb{E}_{\mathbf{x}\sim p} u(\mathbf{x}) + \mathbb{E}_{\mathbf{y}\sim q} v(\mathbf{y}) - \frac{1}{4\epsilon}\mathbb{E}_{\mathbf{x}\sim p,\mathbf{y}\sim q} I(\mathbf{x},\mathbf{y}) \left[(u(\mathbf{x}) + v(\mathbf{y}) - \xi(\mathbf{x},\mathbf{y}))_+\right]^2,$$

For unsupervised OT, $I(\mathbf{x},\mathbf{y}) = 1$ and $\xi(\mathbf{x},\mathbf{y}) = c(\mathbf{x},\mathbf{y})$ For semi-supervised OT, $I(\mathbf{x},\mathbf{y}) = m(\mathbf{x},\mathbf{y})$ and $\xi(\mathbf{x},\mathbf{y}) = g(\mathbf{x},\mathbf{y})$

1.3.3 Solving OT:

u, v are represented by neural networks u_ω, v_ω with parameters ω that are trained by mini-based stochastic optimization algorithms. Using the parameters $\hat{\omega}$ after training, the estimate of optimal transport plan is

$$\hat{\pi}(\mathbf{x},\mathbf{y}) = H(\mathbf{x},\mathbf{y})p(\mathbf{x})q(\mathbf{y}), \text{where } H(\mathbf{x},\mathbf{y}) = \frac{1}{2\epsilon} I(\mathbf{x},\mathbf{y})(u_{\hat{\omega}}(\mathbf{x}) + v_{\hat{\omega}}(\mathbf{y}) - \xi(\mathbf{x},\mathbf{y}))_+$$

1.4 Train with an estimated coupling relationship

1.4.1 Reformulation of paired setting

Let q and p respectively denote the distributions of target data and condition data. In the paired setting, we denote the condition data as $\mathbf{x}_{\mathrm{cond}}(\mathbf{y})$ for a target data $\mathbf{y}$, and p is the measure by push-forwarding q using $\mathbf{x}_{\mathrm{cond}}$, i.e., $p(\mathbf{x}) = \sum_{\{\mathbf{y}:\mathbf{x}_{\mathrm{cond}}(\mathbf{y})=\mathbf{x}\}} q(\mathbf{y})$ over the paired training dataset.

Let $\mathcal{C}(\mathbf{x},\mathbf{y}) = \frac{1}{p(\mathbf{x})}\delta\left(\mathbf{x} - \mathbf{x}_{\mathrm{cond}}(\mathbf{y})\right)$ where δ is the Dirac delta function, then $\mathcal{J}_{\mathrm{DSM}}(\theta)$ can be reformulated as

$$\mathcal{J}_{\mathrm{DSM}}(\theta) = \mathbb{E}_t w_t \mathbb{E}_{\mathbf{x}\sim p}\mathbb{E}_{\mathbf{y}\sim q}\mathcal{C}(\mathbf{x},\mathbf{y})\mathbb{E}_{\mathbf{y}_t\sim p_t|(\mathbf{y}t\mathbf{y})} \left\| s_\theta(\mathbf{y}_t;\mathbf{x},t) - \nabla_{\mathbf{y}_t}\log p_{t|0}(\mathbf{y}_t \mid \mathbf{y})\right\|_2^2.$$

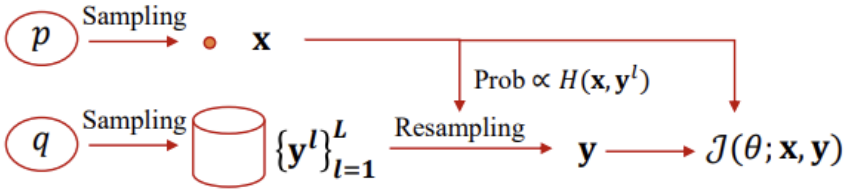
Furthermore, $\gamma(\mathbf{x},\mathbf{y}) = \mathcal{C}(\mathbf{x},\mathbf{y})p(\mathbf{x})q(\mathbf{y})$ is a joint distribution for marginal distributions p and q.

Observations. First, the coupling relationship of condition data and target data is explicitly modeled in $\mathcal{C}(\mathbf{x},\mathbf{y})$ which is for paired $\mathbf{x},\mathbf{y}$.. Second, the joint distribution γ is closely related to the transport plan of L_2-regularized OT.
While for the unpaired or partially paired setting, the definition of $\mathcal{C}(\mathbf{x},\mathbf{y})$ is not obvious due to the lack of paired relationship between $\mathbf{x},\mathbf{y}$. By L_2-regularized OT, we replace $\mathcal{C}(\mathbf{x},\mathbf{y})$ with the compatibility function $H(\mathbf{x},\mathbf{y})$.

1.4.2 Formulation for unpaired and partially paired settings

$$\mathcal{J}_{\mathrm{CDSM}}(\theta) = \mathbb{E}_t w_t \mathrm{E}_{\mathbf{x}\sim p}\mathbb{E}_{\mathbf{y}\sim q} H(\mathbf{x},\mathbf{y})\mathrm{E}_{\mathbf{y}_t\sim p_{t|0}(\mathbf{y}_t|\mathbf{y})} \left\| s_\theta(\mathbf{y}_t;\mathbf{x},t) - \nabla_{\mathbf{y}_t}\log p_{t|0}(\mathbf{y}_t \mid \mathbf{y})\right\|_2^2.$$

OT-guided conditional denoising score matching. H is a "soft" coupling relationship of condition data and target data, because there may exist multiple x satisfying $H(\mathbf{x},\mathbf{y}) > 0$ for each $\mathbf{y}$. While Eq. (8) assumes "hard" coupling relationship, i.e., there is only one condition data $\mathbf{x}$ for each $\mathbf{y}$ satisfying $\mathcal{C}(\mathbf{x},\mathbf{y}) > 0$. We minimize $\mathcal{J}_{\mathrm{CDSM}}(\theta)$ to train the conditional score-based model $s_\theta(\mathbf{y}_t;\mathbf{x},t)$.



1.4.3 Training the Conditional Score-based Model

Past. Using training samples to optimize θ, we can sample **mini-batch data** $\boldsymbol{X}$ and $\boldsymbol{Y}$ from p and q respectively, and then compute $H(\mathbf{x}, \mathbf{y})$ and $\mathcal{J}_{\mathbf{x},\mathbf{y}} = \mathbb{E}_t w_t \mathrm{E}_{\mathbf{y}_t \sim p_t|0(\mathbf{y}t} \mid \mathbf{y})$ $\left\| s_\theta\left(\mathbf{y}_t; \mathbf{x}, t\right) - \nabla_{\mathbf{y}_t} \log p_{t|0}\left(\mathbf{y}_t \mid \mathbf{y}\right)\right\|_2^2$ over the pairs of $(\mathbf{x}, \mathbf{y})$ in $\boldsymbol{X}$ and $\boldsymbol{Y}$.

However, such a strategy is sub-optimal due to sparsity of $H(\mathbf{x}, \mathbf{y})$. To tackle this challenge, we propose a "**resampling-by-compatibility**" strategy to compute. - Sample $\mathbf{x}$ from p and sample $\boldsymbol{Y}_{\mathbf{x}} = \left\{\mathbf{y}^l\right\}_{I=1}^L$ from q; - Resample a $\mathbf{y}$ from $\boldsymbol{Y}_{\mathbf{x}}$ with the probability proportional to $H\left(\mathbf{x}, \mathbf{y}^l\right)$; - Compute the training loss $\mathcal{J}_{\mathbf{x},\mathbf{y}}$ in the above paragraph on the sampled pair $(\mathbf{x}, \mathbf{y})$. (In the applications where we are given training datasets of samples, for each $\mathbf{x}$ in the set of condition data, we choose all the samples $\mathbf{y}$ in the set of target data satisfying $H(\mathbf{x}, \mathbf{y}) > 0$ to construct $\boldsymbol{Y}_{\mathbf{x}}$, and meanwhile store the corresponding values of $H(\mathbf{x}, \mathbf{y})$. This is done before training s_θ. During training, we directly choose $\mathbf{y}$ from $\boldsymbol{Y}_{\mathbf{x}}$ based on the stored values of H, which speeds up the training process.)
After that, we can generate samples by reverse SDE.

1.4.4 Understanding OT-Guided Conditional SBDM

Having generated samples from the conditional transport plan $\hat{\pi}(\mathbf{y} \mid \mathbf{x})$, where $\hat{\pi}(\mathbf{y} \mid \mathbf{x}) = H(\mathbf{x}, \mathbf{y}) q(\mathbf{y})$ is based on $\hat{\pi}(\mathbf{x}, \mathbf{y})$ above.

> For $\mathbf{x} \sim p$, we define the forward SDE $\mathrm{d}\mathbf{y}_t = f\left(\mathbf{y}_t, t\right) \mathrm{d}t + g(t) \mathrm{d}\mathbf{w}$ with $\mathbf{y}_0 \sim \hat{\pi}(\cdot \mid \mathbf{x})$ and $t \in [0, T]$, where f, g, T are given. Let $p_t\left(\mathbf{y}_t \mid \mathbf{x}\right)$ be the corresponding distribution of $\mathbf{y}_t$ and $\mathcal{J}_{\mathrm{CSM}}(\theta) = \mathbb{E}_t w_t \mathbb{E}_{\mathbf{x} \sim p} \mathbb{E}_{\mathbf{y}_t \sim p_t(\mathbf{y}_t|\mathbf{x})} \left\| s_\theta\left(\mathbf{y}_t; \mathbf{x}, t\right) - \nabla_{\mathbf{y}_t} \log p_t\left(\mathbf{y}_t \mid \mathbf{x}\right)\right\|_2^2$, then we have $\nabla_\theta \mathcal{J}_{\mathrm{CDSM}}(\theta) = \nabla_\theta \mathcal{J}_{\mathrm{CSM}}(\theta)$.

Theorem 1 indicates that the trained $s_\theta\left(\mathbf{y}_t; \mathbf{x}, t\right)$ using $\mathcal{J}_{\mathrm{CDSM}}(\theta)$ approximates $\nabla_{\mathbf{y}_t} \log p_t\left(\mathbf{y}_t \mid \mathbf{x}\right)$. Then the **Workflow** breaks down to: (1) Building coupling $\hat{\pi}$ using OT; (2) Sampling clean sample $\mathbf{y}$ from $\hat{\pi}(\mathbf{y} \mid \mathbf{x})$; (3) Adding noise by forward SDE to train s_θ s.t. $s_\theta\left(\mathbf{y}_t; \mathbf{x}, t\right)$ approximates $\nabla_{\mathbf{y}_t} \log p_t\left(\mathbf{y}_t \mid \mathbf{x}\right)$ (4) Generating samples from $\hat{\pi}(\mathbf{y} \mid \mathbf{x})$ by reverse SDE.

1.5 OTCS Realizes Data Transport for Optimal Transport

OTCS offers a diffusion-based approach to transport x to target domain by sampling from optimal conditional transport plan $\pi(\cdot \mid \mathbf{x})$.
$p^{\mathrm{sde}}\left(\cdot \mid \mathbf{x}\right)$ distribution of samples generated by OTCS
$\pi(\mathbf{y} \mid \mathbf{x})$ true conditional optimal transport plan of L_2-regularized OT.
Investigate the upper bound of the expected Wasserstein distance $\mathbb{E}_{\mathbf{x} \sim p} W_2\left(p^{\mathrm{sde}}\left(\cdot \mid \mathbf{x}\right), \pi(\cdot \mid \mathbf{x})\right)$.

Theorem 2. *Suppose the assumptions in Appendix B hold, and $w_t = g(t)^2$, then we have*

$$\mathbb{E}_{\mathbf{x}\sim p} W_2\left(p^{\text{sde}}(\cdot|\mathbf{x}), \pi(\cdot|\mathbf{x})\right) \leq C_1 \left\|\nabla_{\hat{\pi}} \mathcal{L}(\hat{\pi}, u_{\hat{\omega}}, v_{\hat{\omega}})\right\|_1 + \sqrt{C_2 \mathcal{J}_{\text{CSM}}(\hat{\theta})} + C_3 \mathbb{E}_{\mathbf{x}\sim p} W_2\left(p_T(\cdot|\mathbf{x}), p_{\text{prior}}\right), \tag{11}$$

where C_1, C_2, and C_3 are constants to $\hat{\omega}$ and $\hat{\theta}$ given in Appendix B.

Choose SDE to minimize it.

$u_{\hat{\omega}}, v_{\hat{\omega}}$ is near to the saddle point so that gradient norm is minimized.

Training loss as in Theorem 1.

Theorem 2 shows that OTCS can generate samples from $\pi(\cdot\,|\mathbf{x})$.

Denote the Lagrange function for the L_2-regularized unsupervised or semi-supervised OTs as $\mathcal{L}(\pi, u, v)$ with dual variables u, v as follows:

$$\mathcal{L}(\pi, u, v) = \int \left(\xi(\mathbf{x}, \mathbf{y})\pi(\mathbf{x}, \mathbf{y}) + \epsilon \frac{\pi(\mathbf{x}, \mathbf{y})^2}{p(\mathbf{x})q(\mathbf{y})}\right) \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{y} + \int u(\mathbf{x}) \left(\int \pi(\mathbf{x}, \mathbf{y})\mathrm{d}\mathbf{y} - p(\mathbf{x})\right) \mathrm{d}\mathbf{x} + \int v(\mathbf{y}) \left(\int \pi(\mathbf{x}, \mathbf{y})\mathrm{d}\mathbf{x} - q(\mathbf{y})\right) \mathrm{d}\mathbf{y}$$

For semi-supervised OT, π is further constrained by $\pi = m \otimes \tilde{\pi}$.

2 Diffusion model

For conditional sample generation, the conditional score function $\nabla \log p_t(x \mid y)$ needs to be estimated. We slightly abuse the notation to denote s as a conditional score network and $\mathcal{S}$ as the corresponding network class. By introducing an early-stopping time t_0, a conceptual quadratic loss for conditional score estimation is defined as

$$\underset{s\in\mathcal{S}}{\operatorname{argmin}} \int_{t_0}^{T} w(t)\mathbb{E}_{(x_t,y)}\left[\left\|\nabla \log p_t\left(x_t \mid y\right) - s\left(x_t, y, t\right)\right\|_2^2\right] \mathrm{d}t,$$

where $w(t)$ is a time dependent reweighting function. and the equivalence of it is the following implementable loss function,

$$\underset{s\in\mathcal{S}}{\operatorname{argmin}} \int_{t_0}^{T} \mathbb{E}_{(x_0,y)}\left[\mathbb{E}_{x_t\sim \mathrm{N}(\alpha(t)x_0, h(t)I_D)}\left[\left\|\nabla_{x_t} \log \phi_t\left(x_t \mid x_0\right) - s\left(x_t, y, t\right)\right\|_2^2\right]\right] \mathrm{d}t,$$

For any $t \geq 0$, it hold that $\nabla_{x_t} \log p_t\left(x_t \mid y\right) = \nabla_{x_t} \log p_t\left(x_t, y\right)$ since the gradient is taken w.r.t. x_t only. Then plugging in this equation and expanding the norm square on the LHS gives

$$\mathbb{E}_{(x_t,y)\sim P_t}\left[\left\|\nabla_{x_t} \log p_t\left(x_t, y\right) - s\left(x_t, y, t\right)\right\|_2^2\right] = \mathbb{E}_{(x_t,y)\sim P_t}\left[\left\|s\left(x_t, y, t\right)\right\|_2^2 - 2\left\langle \nabla_{x_t} \log p_t\left(x_t, y\right), s\left(x_t, y, t\right)\right\rangle\right] + C.$$

Then it suffices to prove

$$\mathbb{E}_{(x_t,y)\sim P_t}\left[\left\langle \nabla_{x_t} \log p_t\left(x_t, y\right), s\left(x_t, y, t\right)\right\rangle\right] = \mathbb{E}_{(x,y)\sim P_{x\bar{y}}}\mathbb{E}_{x'\sim \mathrm{N}(\alpha(t)x, h(t)I)}\left[\left\langle \nabla_{x'} \phi_t\left(x' \mid x\right), s\left(x', y, t\right)\right\rangle\right]$$

Using integration by parts to rewrite the inner product we have

$$\begin{aligned}
\mathbb{E}_{(x_t,y)\sim P_t}\left[\left\langle \nabla_{x_t} \log p_t\left(x_t, y\right), s\left(x_t, y, t\right)\right\rangle\right] &= \int p_t\left(x_t, y\right)\left\langle \nabla_{x_t} \log p_t\left(x_t, y\right), s\left(x_t, y, t\right)\right\rangle dx_t dy \\
&= \int \left\langle \nabla_{x_t} p_t\left(x_t, y\right), s\left(x_t, y, t\right)\right\rangle dx_t dy \\
&= -\int p_t\left(x_t, y\right) \operatorname{div}\left(s\left(x_t, y, t\right)\right) dx_t dy,
\end{aligned}$$

where denote by $\phi_t\left(x' \mid x\right)$ the density of $\mathrm{N}\left(\alpha(t)x, h(t)I_D\right)$ with $\alpha(t) = \exp(-t/2)$ and $h(t) = 1 - \exp(-t)$, then

$$\begin{aligned}
-\int p_t\left(x_t, y\right) \operatorname{div}\left(s\left(x_t, y, t\right)\right) dx_t dy &= -\mathbb{E}_{(x,y)\sim P_{x\bar{y}}} \int \phi_t\left(x' \mid x\right) \operatorname{div}\left(s\left(x', y, t\right)\right) dx' \\
&= \mathbb{E}_{(x,y)\sim P_{x\bar{y}}} \int \left\langle \nabla_{x'} \phi_t\left(x' \mid x\right), s\left(x', y, t\right)\right\rangle dx' \\
&= \mathbb{E}_{(x,y)\sim P_{x\bar{y}}}\mathbb{E}_{x'\sim \mathrm{N}(\alpha(t)x, h(t)I)}\left[\left\langle \nabla_{x'} \phi_t\left(x' \mid x\right), s\left(x', y, t\right)\right\rangle\right].
\end{aligned}$$

2.1 Classifier and Classifier-Free Guidance

Practical implementations of conditional score estimation, such as classifier and classifier-free guidance methods. Specifically, when conditional information y is discrete, e.g., image

categories, the conditional score $\nabla \log p_t\left(x_t \mid y\right)$ is rewritten via Bayes' rule as

$$\nabla \log p_t\left(x_t \mid y\right)=\nabla \log p_t\left(x_t\right)+\nabla \log c_t\left(y \mid x_t\right),$$

where c_t is the likelihood function of an external classifier. In other words, classifier guidance combines the unconditional score function with the gradient of an external classifier. The external classifier is trained using the diffused data points in the forward process. As a result, the performance of classifier guidance methods is sometimes limited, since it is difficult to train the external classifier with highly corrupted data.

Later, classifier-free guidance proposes to remove the external classifier, circumventing the limitation caused by classifier training. The idea of classifier-free guidance is to introduce a mask signal to randomly ignore y and unifies the learning of conditional and unconditional scores. Specifically, let $\tau \in \{\emptyset$,id $\}$ be a mask signal, where $\emptyset$ means to ignore the conditional information y and id to keep y. Corresponding to the two circumstances, we have

$$\begin{aligned}
\tau=\emptyset: \quad & \int_{t_0}^{T} \mathbb{E}_{(x_0, y)}\left[\mathbb{E}_{x_t \sim \mathrm{N}\left(\alpha(t) x_0, h(t) I_D\right)}\left[\left\|s\left(x_t, \emptyset, t\right)-\nabla_{x_t} \log \phi_t\left(x_t \mid x_0\right)\right\|_2^2\right]\right] \mathrm{d} t \\
\tau=\mathrm{id}: \quad & \int_{t_0}^{T} \mathbb{E}_{(x_0, y)}\left[\mathbb{E}_{x_t \sim \mathrm{N}\left(\alpha(t) x_0, h(t) I_D\right)}\left[\left\|s\left(x_t, y, t\right)-\nabla_{x_t} \log \phi_t\left(x_t \mid x_0\right)\right\|_2^2\right]\right] \mathrm{d} t .
\end{aligned}$$

Therefore,

$$\widehat{s} \in \underset{s \in \mathcal{S}}{\operatorname{argmin}} \int_{t_0}^{T} \mathbb{E}_{(x_0, y)}\left[\mathbb{E}_{\tau \sim P_\tau, x_t \sim \mathrm{N}\left(\alpha(t) x_0, h(t) I_D\right)}\left[\left\|s\left(x_t, \tau y, t\right)-\nabla_{x_t} \log \phi_t\left(x_t \mid x_0\right)\right\|_2^2\right]\right] \mathrm{d} t .$$

Here τ is randomly chosen among $\emptyset$ and id following distribution P_τ. The simplistic choice on P_τ is a uniform distribution on $\{\emptyset$, id $\}$, while it is preferred to bias towards $\tau=$ id in some applications.

References

Gu, Xiang, et al. "Optimal transport-guided conditional score-based diffusion model." *Advances in Neural Information Processing Systems* 36 (2023): 36540-36552.
72603_v3mI6NO.pdf (neurips.cc)
https://fanpu.io/blog/2023/score-based-diffusion-models/
Li, Zihao, et al. "Diffusion model for data-driven black-box optimization." arXiv preprint arXiv:2403.13219 (2024).
Chen, Minshuo, et al. "An overview of diffusion models: Applications, guided generation, statistical rates and optimization." arXiv preprint arXiv:2404.07771 (2024).