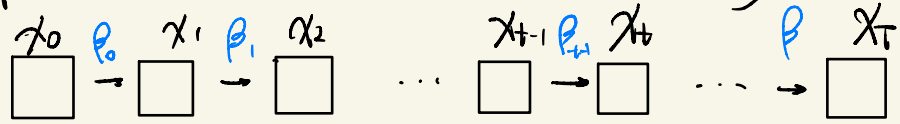
Diffusion models

1. Forward process (deterministic) $q(x_t|x_{t-1})$



$$q(x_t|x_{t-1}) \sim \mathcal{N}(x_t; \sqrt{1-\beta_t}\,x_{t-1}, \beta_t I)$$

Markov process: $q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1})$

one step process: $x_t = \sqrt{1-\beta_t}\,x_{t-1} + \sqrt{\beta_t}\,\varepsilon_{t-1}$

$$= \sqrt{1-\beta_t}\left(\sqrt{1-\beta_{t-1}}\,x_{t-2} + \sqrt{\beta_{t-1}}\,\varepsilon_{t-2}\right) + \beta_t\varepsilon_{t-1}$$

$$= \sqrt{\alpha_t\alpha_{t-1}}\,x_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\,\varepsilon_{t-1}$$

$$= \cdots$$

$$= \boxed{\sqrt{\bar{\alpha}_t}\,x_0 + \sqrt{1-\bar{\alpha}_t}\,\varepsilon}$$

$\alpha_t = 1-\beta_t$

$\beta_T = 1-\alpha_t$

$\sigma^2 = \sigma_1^2 + \sigma_2^2$

continuous form (Stochastic diffusion equation)

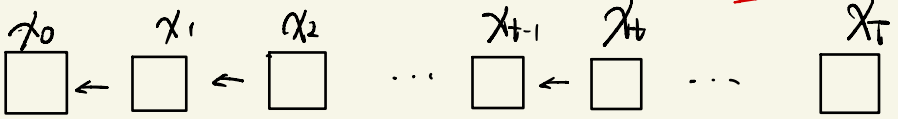
$$x_t = \sqrt{1-\beta_t}\,x_{t-1} + \sqrt{\beta_t}\,\varepsilon_{t-1}$$

$$= \left(1 - \frac{1}{2}\beta_t\right)x_{t-1} + \sqrt{\beta_t}\,\varepsilon_{t-1}$$

$$\Delta x_{t-1} = -\frac{1}{2}\beta_t x_{t-1} + \sqrt{\beta_t}\,\varepsilon_{t-1}$$

$$\frac{dx}{dt} = -\frac{1}{2}\beta_t' x + \sqrt{\beta_t'}\,dw \Leftarrow \frac{dx}{dt} = f(x,t) + g(x,t)\,w$$

Reverse Process $P(x_{t-1}|x_t)$ is unknow → $p_\theta(x_{t-1}|x_t)$



When $\beta_t \ll 1$, $p(x_{t-1}|x_t) \sim \mathcal{N}(x_{t-1}; \mu_t(x_t,t), \Sigma_t(x_t,t))$

Markov Process: $P(x_{0:T}) = P(x_T)\prod_{t=1}^{T} P(x_{t-1}|x_t)$

idea: use a neural network p_θ to approximate $q(x_{t-1}|x_t)$

Sadly: $q(x_{t-1}|x_t)$ is not tractable, trajectories arriving x_{t-1}

$$q(x_{t-1}|x_t) = \frac{q(x_t|x_{t-1})\,q(x_{t-1})}{q(x_t)} \quad \to \times \quad \to \times$$

But: $q(x_{t-1}|x_t, x_0)$ is deterministic!

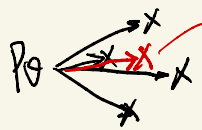
$$q(x_{t-1}|x_t, x_0) = \frac{q(x_t|x_{t-1}, x_0)\,q(x_{t-1}|x_0)}{q(x_t|x_0)}$$

$q(x_t|x_{t-1}, x_0)$: Known; $q(x_{t-1}|x_0)$: Known; $q(x_t|x_0)$: Known

Also: $x_0 = \frac{1}{\sqrt{\bar\alpha_t}}(x_t - \sqrt{1-\bar\alpha_t}\,\varepsilon_t)$

$$\sim \mathcal{N}(x_{t-1}; \tilde\mu_t, \tilde\beta_t I)$$

$$\tilde\mu_t = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar\alpha_t}}\varepsilon_t\right)$$



$\tilde\beta_t$ fixed

Now, p_θ can learn $q(x_{t-1}|x_t)$ in the following way:

$p_\theta(x_t, t) \longrightarrow E_{x_0 \sim p(x_0)}\, q(x_{t-1}|x_t, x_0)$

$p_\theta(x_0, t) \longrightarrow$ $q(x_{t-1}|x_t, x_0)$

$E_{x_t \sim p(x_t|x_0),\, x_0 \sim p(x_0),\, t \sim U(0,T)}$

More on $\tilde{\mu}_t$: $\tilde{\mu}_t(x_t, \epsilon_t) = a_t(x_t - b_t \epsilon_t)$

↑ t Δinput ↑ t

$$\epsilon_t = -\frac{\tilde{\mu}_t}{a_t b_t} + \frac{a_t}{b_t} x_t$$

objective: $E_{x_t, x_0, t} \|\epsilon_\theta - \epsilon_t\|_2^2$

$P_\theta \to \epsilon_\theta \to \epsilon_t$

Idea: Essentially, the network is learning to remove noise. (denoising process)

To derive: use the variational lower bound for $p(x_0)$

Question: How to relate to $\nabla_x \log q(x)$, the score?

Again: $\nabla_x \log q(x)$ is not tractable. But: $\nabla_x \log q(x|x_0)$ is.

$$\nabla_{x_t} \log q(x_t|x_0) = -\nabla_{x_t} \frac{(x_t - \mu)^2}{2\sigma_t^2} = -\frac{2\sigma_t \epsilon_t}{2\sigma_t^2} = -\frac{\epsilon_t}{\sigma_t}$$

$S_\theta \to E_{x_t, x_0, t} \nabla_{x_t} \log q(x_t|x_0) \to \nabla_x \log q(x_t)$

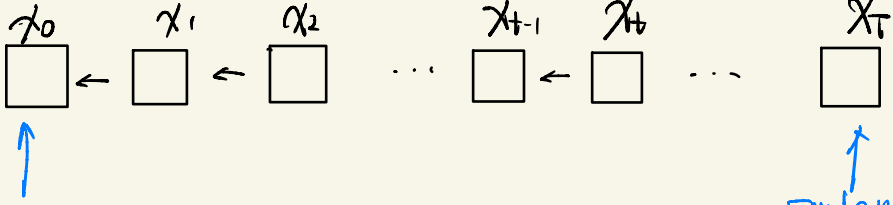
other score matching method:

$\min E_{x_t, t} \|S_\theta(x_t, t)\|_2^2 + \mathrm{tr}\|\nabla_{x_t} S_\theta(x_t, t)\|$ → sliced projected ID

continuous form: (RDE)

$$dx = -\frac{1}{2}\beta_t' x - \beta_t' \nabla_x \log q(x_t) dt + \sqrt{\beta_t'} dw_t$$

Generative model (Denoising diffusion)



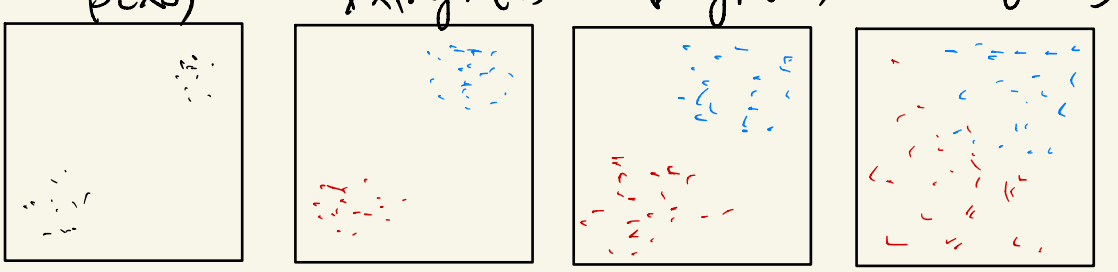
random noise.

$$x_{t-1} = a_t\left[x_t - b_t \varepsilon_\theta(x_t, t)\right] + \sqrt{\beta_t}\,\bar{\varepsilon}$$

$$\vdots$$

$$x_0 = a_1\left[x_1 - b_1 \varepsilon_\theta(x_1, t=1)\right]$$

score-based generative model



$$x_{t-1} = x_t - \lambda \nabla \log p(x_t) + \gamma \bar{\varepsilon}$$

ε_t

avoid local minima.

To derive the objective function: (Maximum Likelihood)

$$\max_\theta P_\theta(x_0) \longrightarrow \max_\theta \log P_\theta(x_0)$$

Variational Lower bound.

$$\log P_\theta(x_0) \geq \log P_\theta(x_0) - KL\left[q(x_{1:T}|x_0) \| P_\theta(x_{1:T}|x_0)\right]$$

$$-\log P_\theta(x_0) \leq -\log P_\theta(x_0) + E_{x_{1:T}\sim q(x_{1:T}|x_0)}\left[\log \frac{q(x_{1:T}|x_0)}{\frac{P_\theta(x_{0:T})}{P_\theta(x_0)}}\right]$$

$$-\log P_\theta(x_0) \leq E_{x_{1:T}\sim q(x_{1:T}|x_0)}\left[\log \frac{q(x_{1:T}|x_0)}{P_\theta(x_{0:T})}\right]$$

$$-E_{x_0\sim q(x_0)}\left[\log P_\theta(x_0)\right] \leq E_{x_{0:T}\sim q(x_{0:T})}\left[\log \frac{q(x_{1:T}|x_0)}{P_\theta(x_{0:T})}\right]$$

$$\min L_0 \longrightarrow \min E_q\left[\log \frac{\prod_{t=1}^{T} q(x_t|x_{t-1})}{P_\theta(x_T)\prod_{t=1}^{T} P_\theta(x_{t-1}|x_t)}\right]$$

$$= \min E_q\left[\log \frac{\prod_{t=1}^{T} q(x_{t-1}|x_t)}{P_\theta(x_T)\prod_{t=1}^{T} P(x_{t-1}|x_t)} \cdot \frac{q(x_t)}{q(x_0)}\right]$$

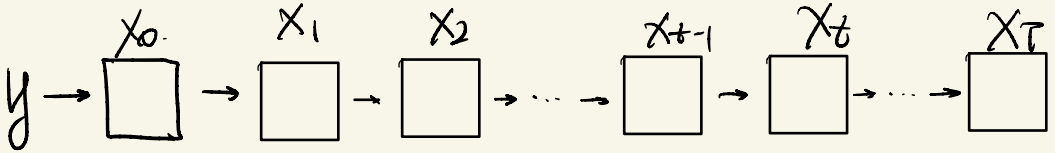
$$= \min E_q\left[\sum_{t=1}^{T}\log \frac{q(x_{t-1}|x_t)}{P_\theta(x_{t-1}|x_t)} + \sum_{t=1}^{T}\log \frac{q(x_t)}{q(x_{t-1})} - \log P_\theta(x_0)\right]$$

$$= \min E_q\left[\log \frac{q(x_T|x_0)}{P_\theta(x_T)} + \sum_{t=1}^{T}\log \frac{q(x_{t-1}|x_t, x_0)}{P_\theta(x_{t-1}|x_t)}\right]$$

Final objective: given t, $x_t \sim q(x_t|x_0)$, $x_0 \sim q(x_0)$,

$$\min_\theta \mathbb{E}_{x_0} KL\left[q(x_{t-1}|x_t, x_0) \,\|\, p_\theta(x_{t-1}|x_t)\right]$$

Conditional diffusion model.



The posterior became: $\log P(x_0|y)$

$$\nabla_{x_0} \log P(x_0|y) = \nabla_{x_0} \log \frac{P(y|x_0)P(x_0)}{P_y}$$

$$= \nabla_{x_0} \log P(y|x_0) + \nabla_{x_0} \log P(x_0)$$

$\nabla_{x_0} \log P(y|x_0)$: classifier / forward model

$\nabla_{x_0} \log P(x_0)$: unconditioned.

$\nabla_x \log P(x|y) \rightarrow$ obj: $\|s_\theta(x_t, t, y) - \nabla_x \log P(x|y)\|_2^2$

$\epsilon_\theta(x_t, x, y) \rightarrow$ obj: $\mathbb{E}_{t, x_t, x_0} \|\epsilon_\theta(x_t, x, y) - \epsilon\|_2^2$