Maximum-Likelihood Training of Score-Based Diffusion Models

Sequentially corrupt data with increasing noise

Learn to reverse corruption to create generative model

Score-Matching with Langevin Dynamics (SMLD) (Song + Ermon 19)

Estimate "score" @ each noise scale

grad of log probability density

Sample from sequence of decreasing noise scales by Langevin Dynamics

Denoising Diffusion Probabilistic Modeling DDPM (Sohl-Dickstein 15, Ho 15)

Can be shown to implicitly compute a score

- score-based models

$x_0 \rightarrow x_T$

$dx = f(x,t)dt + g(t)dw$ [data → noise]

forward SDE

$x_0 \leftarrow x_T$

$dx = [f(x,t) - g^2(t)\nabla_x \log p_t(x)]dt + g(t)dw$ [noise → data]

reverse SDE

Perturb data with a continuum of distributions that evolve in time according to diffusion process

Diffuse data to noise with non-parametric SDE

The reverse process can be ~~solved~~ described by reverse-time SDE that can be derived from forward SDE given $\nabla_x \log p_t(x)$

Score-based diffusion can be converted to flow for numerical likelihood computation

Let $p(x)$ unknown distribution of D-dim iid samples

Score-based diff models diffuse $p(x) \to$ noise by SDE

$$dx = f(x,t)dt + g(t)dw \quad \leftarrow \text{infinitesimal Gaussian noise}$$

where $f(\cdot,t)$ drift coefficient

$g(t) \in \mathbb{R}$ diffusion coefficient

$w \in \mathbb{R}^D$ Wiener process (Brownian motion) aka diffusion by Langevin eq

$W_0 = 0$
independent increments
$W_{t+u} - W_t$ Gaussian $(0, u)$
W_t continuous in t

$x(t)$ then is a diffusion process

$p_t(x)$ marginal distribution

$p_{0t}(x'|x)$ transition distribution from x_0 to x_t

Intuitively: smoothen the data distribution in time (by adding noise)

We smooth into an analytically tractable prior $\pi(x)$ at $t = T$

i.e. $p_T(x) \approx \pi(x)$

this can be done by a family of SDEs

Now we can reverse the SDE

$$dx = f(x,t)dt + g(t)dw$$

data $x(0)$ → $x(T)$ noise

$$dx = \left[f(x,t) - g(t)^2 \nabla_x \log p_t(x)\right]dt + g(t)d\bar{w}$$

Choose f and g so we have reversal by

this is the learnable piece

$\bar{w}$: standard Wiener process

Fit a "score-based model" $s_\theta(x,t) \approx \nabla_x \log p_t(x)$ [notice not normalized]

$$J_{SM}(\theta; \lambda(\cdot)) := \frac{1}{2}\int_0^T \mathbb{E}_{p_t(x)}\left[\lambda(t)\left\| \nabla_x \log p_t(x) - s_\theta(x,t)\right\|_2^2\right]dt$$ loss function

$\lambda : [0, T] \to \mathbb{R}^+$ the "score matching objective"

We want to transform this objective to something computable

By denoising score-matching [Vincent MLSS 11] we can get J up to a constant indep of θ

$$J_{DSM}(\theta;\lambda(\cdot)) := \frac{1}{2}\int_0^T \mathbb{E}_{p(x)p_{0t}(x'|x)}\left[\lambda(t)\left\|\nabla_{x'}\log p_{0t}(x'|x) - s_\theta(x',t)\right\|_2^2\right]dt$$

when $f_\theta(x,t)$ linear in x [look eq]

$p_{0t}(x'|x)$ is Gaussian

To estimate draw (t, x, x')

$t \sim [0,T]$

$x \sim p(x)$ data sample

$x' \sim p_{0t}(x'|x) \Rightarrow \nabla_{x'}\log p_{0t}(x'|x)$ since p gaussian

Once we have $s_\theta(x,t)$ we use numerical SDE solvers to sample

Likelihood of score-based diffusion Models

There are 2 probabilistic models for which we can define a likelihood

1. $p_\theta^{SDE}(x)$

Let $\{\hat{x}_\theta(t)\}_{t\in[0,T]}$ a stochastic process given by

$$d\hat{x} = \left[f(\hat{x},t) - g(t)^2 s_\theta(\hat{x},t)\right]dt + g(t)\,d\bar{w} \qquad \hat{x}_\theta(T) \sim \pi$$

Then $p_\theta^{SDE}(x)$ is marginal distribution of $\hat{x}_\theta(0)$

We sample by

$\hat{x}_\theta(0) \sim p_\theta^{SDE}$ numerically solving reverse-time SDE w/ noise $x_\theta(T)\sim\pi$

2. $p_\theta^{ODE}(x)$ derived from SDE's probability flow ODE

Just ODE with the same $p_t(t)$ given by

$$\frac{dx}{dt} = f(x,t) - \frac{1}{2}g(t)^2\nabla_x \log p_t(x) \quad \text{AND score function}$$

$$\approx f(x,t) - \frac{1}{2}g(t)^2 s_\theta(\bar{x},t)$$

this is actually a CNF!

So given prior $\pi(x)$ and trajectory functions $\tilde{x}_\theta : [0,T] \to \mathbb{R}^n$

satisfying ODE then p_θ^{ODE} is the marginal of $\tilde{x}_\theta(0)$ when $\tilde{x}_\theta(T) \sim \pi$

Can evaluate exactly with ODE solver

Computing this ODE $\log p_\theta^{ODE}(x)$ is tractible

BUT solving requires running a solver at every step

Similarly we can't evaluate p_θ^{SDE} but we can solve for lower bound.

Upper
~~Lower~~ Bounds

With correct choice of λ

$J_{SM}(\theta; \lambda(\cdot))$ becomes a ~~lower~~ upper bound $D_{KL}(p \| p_\theta^{SDE})$

when $\lambda(t) = g(t)^2$

$$\Rightarrow D_{KL}(p \| p_\theta^{SDE}) \leq J_{SM}(\theta; g(\cdot)^2) + D_{KL}(p_T \| \pi)$$

AND

$$-\mathop{\mathbb{E}}_{p(x)}[\log p_\theta^{SDE}(x)] \leq J_{SM}(\theta; g(\cdot)^2) + C_1 = J_{DSM}(\theta; g(\cdot)^2) + C_2$$

AND when $s_\theta(x,t) = \nabla_x \log q_t(x) \quad \forall t \in [0,T]$ then

$p_\theta^{SDE} = p_\theta^{ODE} = q$ and

$$D_{KL}(p \| p_\theta^{SDE}) = J_{SM}(\theta; g(\cdot)^2) + D_{KL}(p_T \| \pi)$$