Score based generative modeling

A Preprint

Whole idea comes from 2 previously presented methods: Score matching with Langevin dynamics (SMLD) and Denoise diffusion probabilistic model. SMLD idea is to estimate score(gradient of the log prob of data distribtuion) at different noise levels(different levels of gaussian noise are injected). DDPM however trains a sequence of probabilistic models to reverse each step of the noise corruption, using knowledge of the functional form of the reverse distributions to make training tractable. For continuous state spaces, the DDPM training objective implicitly computes scores at each noise scale.

- Any general-purpose SDE solver to integrate the reverse-time SDE for sampling can be employed. It is important to note that general methods like the Euler-Maruyama method or Stochastic Runge-Kutta methods at each step introduces a small amount of numerical error. Over thousands of steps, this error can accumulate, causing the final generated sample to be slightly off from the true data distribution. Due to this problem, authors introduce Predictor-Corrector (PC) Samplers and The Probability Flow (PF) ODE Sampler.
- Controllable generation: as big model for score estimation never changes, authors state the fact that by changing A small, task-specific "guidance"model that provides the gradient guidance can be achieved in any domain
- Unified framework

$p_\sigma(\hat{x}|x) \triangleq \mathcal{N}(\hat{x}; x; \sigma^2 I)$ – a perturbation kernel (just a way to add noise to data);

$$p_\sigma(\hat{x}|x) \triangleq \mathcal{N}(\hat{x}; x; \sigma^2 I) \text{ – a perturbation kernel (just a way to add noise to data)} \tag{1}$$

$$p_\sigma(\hat{x}) = \int p_{data}(x) p_\sigma(\hat{x}|x) dx. \tag{2}$$

Noise scales: $\sigma_1 < \sigma_2 \ldots < \sigma_{N-1} < \sigma_N$. $\sigma_1 = \sigma_{\min}$ is that min, that $p_{\sigma_{\min}}(x) = p_{data}(x)$.
$\sigma_{\max} : p_{\sigma_{\max}}(x) \approx \mathcal{N}\left(x; 0, \sigma_{\max}^2 I\right)$

$$\theta^* = \arg\min_\theta \sum_{i=1}^{N} \sigma_i^2 \, \mathbb{E}_{x \sim p_{\text{data}}(x)} \, \mathbb{E}_{\tilde{x} \sim p_{\sigma_i}(\tilde{x}|x)} \left[\| s_\theta(\tilde{x}, \sigma_i) - \nabla_{\tilde{x}} \log p_{\sigma_i}(\tilde{x} \mid x) \|_2^2 \right] \tag{1}$$

Given sufficient data and model capacity, the optimal score-based model $s_{\theta^*}(x, \sigma)$ matches $\nabla_x \log p_\sigma(x)$ almost everywhere for $\sigma \in \{\sigma_i\}_{i=1}^N$.

For sampling run M steps of Langevin MCMC to get a sample from each $p_{\sigma_i}(x)$ sequentially:

$$x_i^m = x_i^{m-1} + \epsilon_i \, s_{\theta^*}(x_i^{m-1}, \sigma_i) + \sqrt{2\epsilon_i} \, z_i^m, \quad m = 1, 2, \ldots, M \tag{2}$$

where $\epsilon_i > 0$ is the step size, and $z_i^m \sim \mathcal{N}(0, I)$ is standard Gaussian noise.

This is repeated for $i = N, N-1, \ldots, 1$ with initialization:

$$x_N^0 \sim \mathcal{N}(0, \sigma_{\max}^2 I), \quad x_i^0 = x_{i+1}^M \text{ for } i < N$$

As $M \to \infty$ and $\epsilon_i \to 0$, the final sample x_1^M becomes an exact sample from $p_{\sigma_{\min}}(x) \approx p_{\text{data}}(x)$ under regularity conditions.

Consider a sequence of positive noise levels $0 < \beta_1, \beta_2, \ldots, \beta_N < 1$. For each data point $x_0 \sim p_{\text{data}}(x)$, we define a discrete Markov chain $\{x_0, x_1, \ldots, x_N\}$ such that:

$$p(x_i \mid x_{i-1}) = \mathcal{N}(x_i; \sqrt{1-\beta_i}\, x_{i-1}, \beta_i I),$$

which implies that the marginal distribution of x_i given the original x_0 is:

$$p_{\alpha_i}(x_i \mid x_0) = \mathcal{N}(x_i; \sqrt{\alpha_i}\, x_0, (1-\alpha_i) I),$$

where $\alpha_i := \prod_{j=1}^{i}(1-\beta_j)$.

Similar to SMLD (Score-Matching with Langevin Dynamics), the perturbed data distribution is defined as:

$$p_{\alpha_i}(\tilde{x}) := \int p_{\text{data}}(x)\, p_{\alpha_i}(\tilde{x} \mid x)\, dx.$$

The noise schedule is chosen such that $x_N \approx \mathcal{N}(0, I)$.

We define a variational reverse Markov chain parameterized as:

$$p_\theta(x_{i-1} \mid x_i) = \mathcal{N}(x_{i-1}; \frac{1}{\sqrt{1-\beta_i}}(x_i + \beta_i s_\theta(x_i, i)), \beta_i I),$$

which is trained using a reweighted denoising score-matching objective (a variant of the ELBO):

$$\theta^* = \arg\min_\theta \sum_{i=1}^{N} (1-\alpha_i)\, \mathbb{E}_{x \sim p_{\text{data}}}\, \mathbb{E}_{\tilde{x} \sim p_{\alpha_i}(\tilde{x}|x)} \left[\| s_\theta(\tilde{x}, i) - \nabla_{\tilde{x}} \log p_{\alpha_i}(\tilde{x} \mid x) \|_2^2 \right]. \tag{3}$$

After training, samples can be generated by initializing:

$$x_N \sim \mathcal{N}(0, I)$$

and recursively applying the reverse Markov process:

$$x_{i-1} = \frac{1}{\sqrt{1-\beta_i}} \left(x_i + \beta_i s_{\theta^*}(x_i, i) \right) + \sqrt{\beta_i}\, z_i, \quad z_i \sim \mathcal{N}(0, I), \quad i = N, \ldots, 1. \tag{4}$$

This sampling process is called ancestral sampling, as it amounts to sampling from the full generative model:

$$\prod_{i=1}^{N} p_\theta(x_{i-1} \mid x_i).$$

The loss in Eq. (3) corresponds to L_{simple}, and is structurally similar to Eq. (1): both are weighted sums of denoising score matching objectives. The optimal model $s_{\theta^*}(\tilde{x}, i)$ matches the score of the perturbed data distribution $\nabla_x \log p_{\alpha_i}(x)$. Notably, the weights σ_i^2 in Eq. (1) and $1-\alpha_i$ in Eq. (3) are functionally related to the norms of the score gradients:

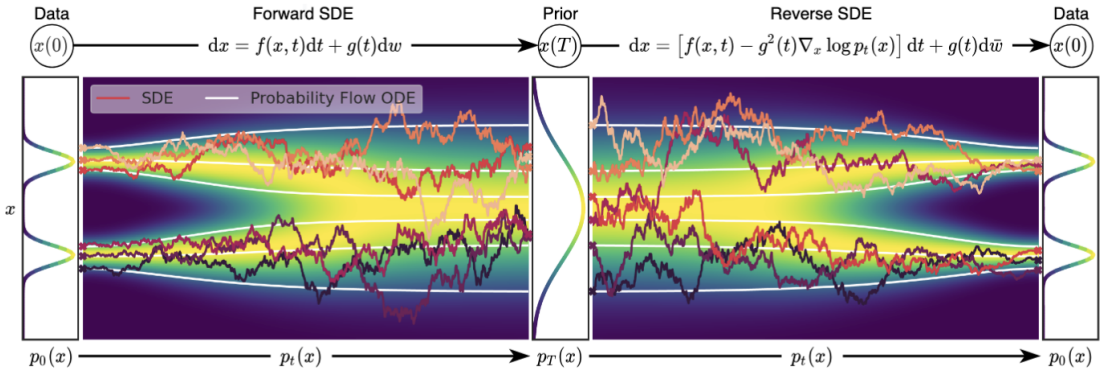
$$\sigma_i^2 \propto \frac{1}{\mathbb{E}[\|\nabla_x \log p_{\sigma_i}(\tilde{x} \mid x)\|_2^2]}, \quad 1-\alpha_i \propto \frac{1}{\mathbb{E}[\|\nabla_x \log p_{\alpha_i}(\tilde{x} \mid x)\|_2^2]}.$$

This diffusion process can be modeled as the solution to an It^o SDE:

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

where $g(t)dw$ represents an infinitesimal step of a Wiener process (or Brownian motion). You can think of it as a tiny, completely random kick from a Gaussian distribution at every single moment in time. This is where the noise is injected; $f(*;t)$ is a vector function called drift coefficient of x(t); diffusion coefficient of x(t) is g(t) and assumed to be scalar. Such SDE has a solution if all coefficients are globally Lipschitz in both state and time.

Reminder: Lipcshitz means $|f(x_1) - f(x_2)| \leq K|x_1 - x_2|$(The function can't have vertical slopes or regions where it suddenly becomes infinitely steep.)



Reversing: $x(0)$ can be obtained from $x(T) \sim p_T$ by modeling reverse process:

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

To estimate the data score function $\nabla_x \log p_t(x)$, the authors propose to train a time-dependent score model $s_\theta(x,t)$ using a continuous generalization of the discrete objectives in Eqs. (1) and (3). The training objective is defined as:

$$\theta^* = \arg\min_\theta \mathbb{E}_t \left[\lambda(t)\, \mathbb{E}_{x^{(0)}\sim p_0}\, \mathbb{E}_{x^{(t)}\sim p_{0t}(x^{(t)}|x^{(0)})} \left\| s_\theta(x^{(t)},t) - \nabla_{x^{(t)}} \log p_{0t}(x^{(t)} \mid x^{(0)}) \right\|_2^2 \right] \tag{7}$$

Here, $\lambda : [0,T] \to \mathbb{R}_{>0}$ is a positive weighting function, and t is sampled uniformly from the interval $[0,T]$. The sample $x^{(0)} \sim p_0(x)$ is a clean data point, and $x^{(t)} \sim p_{0t}(x^{(t)} \mid x^{(0)})$ is a noised version of it at time t.

Under suitable assumptions on data and model capacity, the optimal solution $s_{\theta^*}(x,t)$ converges to the true score function $\nabla_x \log p_t(x)$ for almost all x and t.

Similar to SMLD and DDPM, an appropriate choice of weighting function is $\lambda(t) \propto 1/\mathbb{E}[\|\nabla_{x^{(t)}} \log p_{0t}(x^{(t)} \mid x^{(0)})\|_2^2]$.

Efficient optimization of Eq. (7) generally requires access to the transition kernel $p_{0t}(x^{(t)} \mid x^{(0)})$. For affine drift functions $f(\cdot,t)$, this kernel corresponds to a Gaussian distribution with known closed-form expressions for the mean and covariance. In the case of more general stochastic differential equations (SDEs), the kernel can be obtained by solving the Kolmogorov forward equation or by simulating the SDE and replacing denoising score matching with sliced score matching, thereby avoiding the need to explicitly compute score gradients.

Continuous-Time Score-Based Generative Modeling Summary

SDE View of Diffusion Models. SMLD and DDPM correspond to discretizations of continuous-time SDEs. In the limit as the number of steps $N \to \infty$, the discrete noise schedules $\{\sigma_i\}$ and $\{\beta_i\}$ become continuous functions of time, leading to stochastic processes governed by Itô SDEs.

- VE SDE: Exploding-variance process

$$dx = \sqrt{\frac{d\sigma^2(t)}{dt}}\, dw$$

- VP SDE: Variance-preserving process

$$dx = -\frac{1}{2}\beta(t)x\, dt + \sqrt{\beta(t)}\, dw$$

- sub-VP SDE: Tighter variance control

$$dx = -\frac{1}{2}\beta(t)x\, dt + \sqrt{\beta(t)(1 - e^{-2\int_0^t \beta(s)ds})}\, dw$$

Reverse SDE Sampling. Once the score model $s_\theta(x, t)$ is trained, sampling is performed by solving the reverse-time SDE using numerical solvers. Several options include:

- Ancestral Sampling: A discretization of the reverse VP SDE.
- Reverse Diffusion Sampler: Matches forward and reverse discretizations.
- Predictor-Corrector (PC): Alternates between numerical prediction and score-based MCMC correction (e.g., Langevin steps).

Probability Flow ODE. Every diffusion SDE has a deterministic counterpart:

$$dx = \left(f(x, t) - \frac{1}{2} g(t)^2 \nabla_x \log p_t(x) \right) dt$$

This ODE shares the same marginals as the SDE and can be used for likelihood estimation or fast sampling.

Exact Likelihoods. The ODE formulation enables exact log-likelihood computation via the instantaneous change-of-variables formula, making the model not only generative but also density-estimating.

Latent Representation. By integrating the forward ODE from $x^{(0)}$ to $x^{(T)}$, each datapoint has a uniquely defined latent code. This allows editing and interpolation in latent space.

Sampling Efficiency. Using adaptive solvers, high-quality samples can be generated with far fewer score evaluations by trading off numerical precision.