

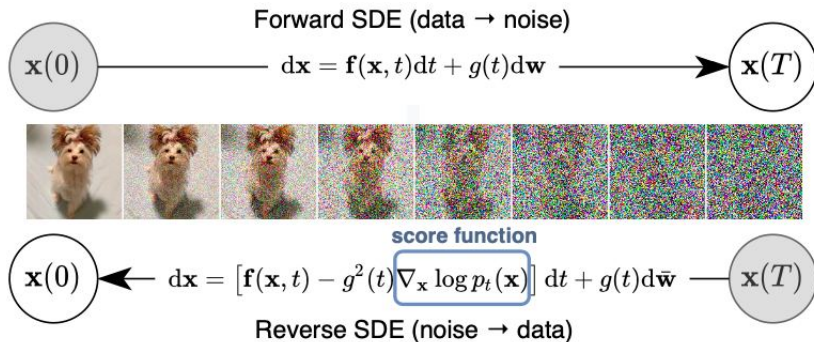
The Hidden Linear Structure in Score-Based Models and its Application

Daniil Shlenskii, Kseniia Petrushina, Arina Chumachenko, Bair
Mikhailov, Nikita Kornilov

Skoltech

December 19, 2023

Score-based models intro



Score-based models intro

Goal

We want to sample from $p_{real}(\mathbf{x})$ which we don't know

Noising process (Forward process)

We set parameters of Forward SDE in a way, that marginal distributions have the following form.

$$p(\mathbf{x}, \sigma_t) = p_{real}(\mathbf{x}) * \mathcal{N}(\mathbf{x}|0, \sigma_t^2 I)$$

Denoising process(Backward process)

In that setup we can sample images using the following ODE which the same marginal distributions as aforementioned Reverse SDE.

$$d\mathbf{x} = -\dot{\sigma}_t \sigma_t \nabla_{\mathbf{x}} \log p(\mathbf{x}, \sigma_t) dt \quad (1)$$

Score function

We parameterize our score model in the following way:

$$\nabla_{\mathbf{x}} \log p(\mathbf{x}, \sigma) \approx s_{\theta}(\mathbf{x}, \sigma) = \frac{D_{\theta}(\mathbf{x}, \sigma) - \mathbf{x}}{\sigma},$$

Where $D_{\theta}(\mathbf{x}, \sigma)$ is a prediction of a corresponding denoised object given σ and $\mathbf{x} \sim p(\mathbf{x}, \sigma)$

Normal case

Analytical form

Let real data $p_{real}(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\mu, \Sigma)$, then we have $p(\mathbf{x}, \sigma) = \mathcal{N}(\mathbf{x}|\mu, \Sigma + \sigma^2 I)$ and

$$\nabla_{\mathbf{x}} \log p(\mathbf{x}, \sigma) = (\sigma^2 I + \Sigma)^{-1}(\mu - \mathbf{x})$$

Numerical linear algebra

Let positively semidefined Σ has rank r and by SVD decomposition $\Sigma = U\Lambda U^\top$, where orthogonal $U = [\mathbf{u}_1, \dots, \mathbf{u}_r]$ and $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_r)$. This form and NLA help us to compute efficiently

$$(\sigma^2 I + \Sigma)^{-1} = \frac{1}{\sigma^2} (I + U\Lambda U^\top / \sigma^2)^{-1}$$

Woodbury matrix inversion

$$(I + AB)^{-1} = I - A(I + BA)^{-1}B$$

With $A = U$, $B = \Lambda U^\top / \sigma^2$ and orthogonal $U^\top U = I_r$ we get

$$\begin{aligned}(I + U\Lambda U^\top / \sigma^2)^{-1} &= (I - U\tilde{\Lambda}_\sigma U^\top) \\ \tilde{\Lambda}_\sigma &= \text{diag} \left[\frac{\lambda_k}{\lambda_k + \sigma^2} \right]\end{aligned}$$

Finally,

$$\nabla_{\mathbf{x}} \log p(\mathbf{x}, \sigma) = \frac{1}{\sigma^2} (I - U\tilde{\Lambda}_\sigma U^\top)(\mu - \mathbf{x})$$

Solving ODE (1)

$$d\mathbf{x} = -\sigma \nabla_{\mathbf{x}} \log p(\mathbf{x}, \sigma) d\sigma$$

$$d\mathbf{x} = \frac{1}{\sigma} (I - U \tilde{\Lambda}_{\sigma} U^{\top}) (\mathbf{x} - \mu) d\sigma$$

Let's find solution in basis U , i.e. $\mathbf{x} = \mu + \sum_{k=1}^r c_k(\sigma) \mathbf{u}_k$, where projection $c_k(\sigma) = \mathbf{u}_k^{\top} (\mathbf{x}(\sigma) - \mu)$. Then solve it for each k separately

$$\frac{1}{\sigma} \left(1 - \frac{\lambda_k}{\lambda_k + \sigma^2} \right) = \frac{\sigma}{\lambda_k + \sigma^2} \implies dc_k(\sigma) = \frac{\sigma}{\lambda_k + \sigma^2} c_k(\sigma) d\sigma$$

Solving ODE (1)

$$\begin{aligned}dc_k(\sigma) &= \frac{\sigma}{\lambda_k + \sigma^2} c_k(\sigma) d\sigma \\d \log c_k(\sigma) &= d \log \sqrt{\lambda_k + \sigma^2} \\c_k(\sigma) &= K \sqrt{\lambda_k + \sigma^2}\end{aligned}$$

Usually σ_T^2 is very large and $\mathcal{N}(\mathbf{x}|\mu, \sigma_T^2 I + \Sigma) \approx \mathcal{N}(0, \sigma_T^2 I)$. With initial condition $\mathbf{x}(T) = \mathbf{x}_T \sim \mathcal{N}(0, \sigma_T^2 I)$ we have

$$c_k(\sigma_t) = \sqrt{\frac{\lambda_k + \sigma_t^2}{\lambda_k + \sigma_T^2}} \mathbf{u}_k^\top (\mathbf{x}_T - \mu)$$

Analytical solution

$$\mathbf{x}_t = \mu + \frac{\sigma_t}{\sigma_T} (I - UU^\top)(\mathbf{x}_T - \mu) + \sum_{k=1}^r \sqrt{\frac{\lambda_k + \sigma_t^2}{\lambda_k + \sigma_T^2}} \mathbf{u}_k \mathbf{u}_k^\top (\mathbf{x}_T - \mu)$$

Additional second term is off-manifold term in case if U is rank deficient.

Score function

Instead of neural network $D_\theta(\mathbf{x}, \sigma)$ we have analytical solution

$$\begin{aligned} D(\mathbf{x}_t, \sigma_t) &= \mathbf{x}_t + \frac{1}{\sigma_t^2} (I - U\tilde{\Lambda}_{\sigma_t}U^\top)(\mu - \mathbf{x}_t) \\ &= \mu + \sum_{k=1}^r \frac{\lambda_k}{\sqrt{(\lambda_k + \sigma_t^2)(\lambda_k + \sigma_T^2)}} \mathbf{u}_k \mathbf{u}_k^\top (\mathbf{x}_T - \mu) \end{aligned}$$

Experiments Idea

Problem

We need to do many steps with $D_\theta(\mathbf{x}, \sigma)$ to get proper sample from ODE (1)

Empirical solution

If $p_{real}(\mathbf{x})$ is not normal we still can calculate mean μ and covariance Σ from data points. The Gaussian analytical solution provides a surprisingly good approximation to the early sampling trajectory. We skip several initial steps and start with

$$\mathbf{x}_t = \mu + \frac{\sigma_t}{\sigma_T}(I - UU^\top)(\mathbf{x}_T - \mu) + \sum_{k=1}^r c_k(\sigma_t)\mathbf{u}_k$$

and then continue with trained $D_\theta(\mathbf{x}_t, \sigma_t)$

Results of the experiment

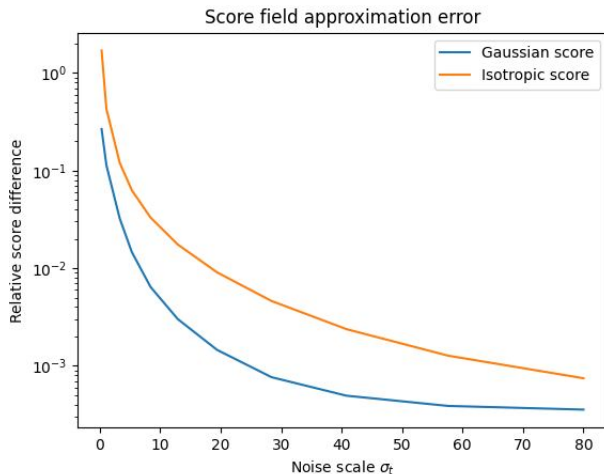


Figure 1: Approximation error of score learned by score neural network with analytical approximations

Results of the experiment

| #(skipped steps) | FID |
|------------------|---------|
| 0 | 1.959 |
| 1 | 1.957 |
| 2 | 1.95 |
| 3 | 1.942 |
| 4 | 1.934 |
| 5 | 1.935 |
| 6 | 1.965 |
| 7 | 2.127 |
| 8 | 3.16 |
| 10 | 23.532 |
| 12 | 108.217 |

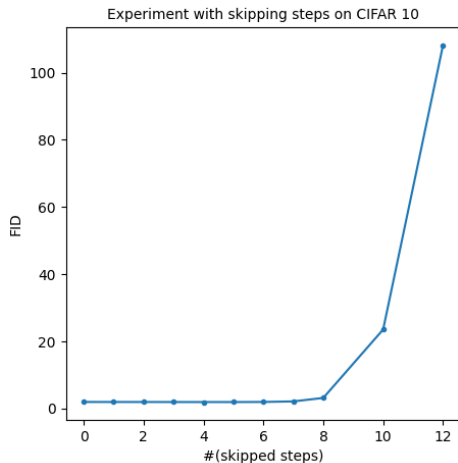


Figure 2: Image quality (FID score) of the method as a function of skipped steps on CIFAR10

Examples



Figure 3: Sampled image as a function of skipped steps (from 18 steps)



Wang, Binxu, and John J. Vastola

The Hidden Linear Structure in Score-Based Models and its Application

arXiv preprint arXiv:2311.10892 (2023).

Our Roles and Links

- Daniil and Kseniia – Code
- Bair, Arina, Nikita – Presentation and README
- Our GitHub
`https://github.com/pkseniya/
TheHiddenLinearStructureInScore-BasedModels`