# @hoDenoisingDiffusionProbabilistic2020

# Denoising Diffusion Probabilistic Models

Jonathan Ho, Ajay Jain, Pieter Abbeel
2020
http://arxiv.org/abs/2006.11239
Zotero
Tags:

- [x] ~~handwritten notes created~~

## Notes

### Summary

Denoising score matching aims to minimize the KL-divergence between joint forward PDF $Q(x_{0:T})$ and joint backward PDF $P_\theta(x_{0:T})$. To do so, the learning process dramatically prioritizes the reverse sampling at low noise levels, with time slices weight $\lambda(t) = \frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}$. DDPM, however, sets $\lambda(t) = constant$. The resulting loss function is mathematically simpler, as well as empirically superior due to better quality of generated images.

### Forward Process

DDPM analysis applies only to variance-preserving (@mengSDEditGuidedImage2022) type of noise. Each forward step is parametrized by:

$$x_t = \sqrt{1-\beta_t}\, x_{t-1} + \sqrt{\beta_t}\, \mathbf{z} \tag{1}$$

or

$$q(x_t|x_{t-1}) = N(x_t; \sqrt{1-\beta_t}\, x_{t-1}, \beta_t) \tag{1.1}$$

Where $\mathbf{z} \sim N(0,1)$ and $\beta_1 \dots \beta_T$ are prescheduled variances. Recall that the solution to Orstein-Uhlenbeck process $dx = -x dt + \sqrt{2}\, dW$ is $x(t) = \mathbf{x(0)}e^{-t} + \mathbf{z}\sqrt{1-e^{-2t}}$. Therefore, as long as equation (1) is satisfied, each value $\beta_t$ represents a **time interval** during an Orstein-Uhlenbeck process. $\beta_t$ can be a constant.

Knowing $x_0$, any arbitrary $x_t$ can be **sampled directly in one step** (with $\alpha_t = 1-\beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$ ):

$$< x_t >= \sqrt{\prod_{t=1}^T 1-\beta_t}\, x_0 = \sqrt{\bar{\alpha}_t}\, x_0$$

Obtained by exponentially decaying the mean by applying (1) recursively.

$$Var[x_t] = 1 - \bar{\alpha}_t$$

Using the (trivial) fact that (1) hold for any t.

In other words, we can perform **one-step forward sampling**:

$$q(x_t|x_0) = N(x_t; \sqrt{\bar{\alpha}_t}\, x_0, (1-\bar{\alpha}_t)\mathbf{I})$$

## Reverse Process

The reverse process is parametrized by one single neural network, conditioned on noise level, t.

$$P_\theta(x_{t-1}|x_t)$$

Joint forward and reverse PDF

$$Q(x_{0:T}) = \prod_{t=1}^{T} q_(x_t|x_{t-1})$$
$$P_\theta(x_{0:T}) = x_T \prod_{t=1}^{T} P_\theta(x_{t-1}|x_t)$$

Training process is to minimize

$$KL(Q(x_{0:T})||P_\theta(x_{0:T})) = \mathbb{E}_{x\sim Q}[log\frac{Q(x)}{P(x)}] = -\mathbb{E}_{x\sim Q}[log\frac{P(x)}{Q(x)}]$$

Or to maximize

$$ELBO(P_\theta) = \mathbb{E}_{x\sim Q}[log\frac{P(x)}{Q(x)}] = \mathbb{E}_Q[\sum_{t=1}^{T} log\frac{p_\theta(x_{t-1}|x_t)}{q(x_t|x_{t-1})}]$$

Where $ELBO$ is a functional of $P_\theta$ **(need to double check the definition of ELBO)**
Let's parametrize

$$p_\theta(x_{t-1}|x_t) = N(x_{t-1}; \mu_\theta(x_t,t), \sigma_t)$$
$$\text{where } \mu_\theta = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t,t))$$

where $\sigma_t$ is step-wise noise in the reverse process, a hyper-parameter set empirically. A choice consistent with reverse SDE [1] is to match the variance of $p_\theta(x_{t-1}|x_t)$ and that of $q(x_t|x_{t-1})$ , and set $\sigma_t = \sqrt{\beta_t}$. The KL-divergence of two gaussians of same width has simple expressions. Hence the learning Loss can be simplified:

$$Loss(P_\theta) = \mathbb{E}_{t\sim[1,T], x_0\sim data, \epsilon\sim N(0,1)}[\lambda(t)||\epsilon - \epsilon_\theta(x_t,t)||^2]$$

where

$$x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\mathbf{z}$$

$$\lambda(t) = \frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\bar{\alpha}_t)}$$

Empirically, $\lambda(t) = 1$ results in better learning quality.

## Algorithm: training and sampling

**Algorithm 1** Training

1: **repeat**
2: $\quad \mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $\quad t \sim \mathrm{Uniform}(\{1, \dots, T\})$
4: $\quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: $\quad$ Take gradient descent step on
$\qquad \nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$
6: **until** converged

**Algorithm 2** Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \dots, 1$ **do**
3: $\quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\quad \mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$

## Connection to Score Matching SDE (Score SDE)[2]

**DDPM and SDE differ only in the weights of time-slices:** $\lambda(t)$

Definition: Score approximates the gradient of log-likelihood

$$S_\theta(x_t, t) = \frac{1}{P(x_t)} \nabla_{x_t} P(x_t)$$

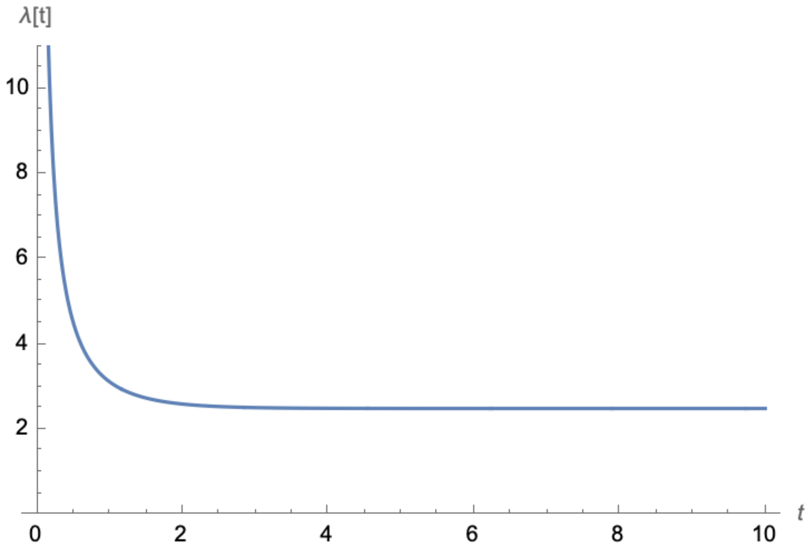
The learning loss of score is [2-1]:

$$Loss = \mathbb{E}_{t\sim[1,T], x_0\sim data, \epsilon\sim N(0,1)}[\lambda(t)||\epsilon + \sqrt{1-\bar{\alpha}_t} S_\theta(x_t, t)||^2]$$

where

$$\lambda(t) = \frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\bar{\alpha}_t)}$$

Where $\sigma_t^2$ is the variance of step-wise noise added during reversal. A choice consistent with reverse SDE is $\sigma_t = \sqrt{\beta_t}$. [3]




Score training process dramatically prioritizes the accuracy of reverse sampling at low noise levels. By setting $\lambda(t) = constant$, DDPM effectively increases the reversal accuracy at high noise levels, which increases the performance for image generation.

## Connection to continuous SDE

The Orstein-Uhlenbeck process is described by SDE

$$dx = -Kxdt + \sigma\, dW$$

The corresponding Fokker-Planck is (@StochasticMethodsSpringerLink)

$$\frac{\partial(p(x,t))}{\partial t} = \frac{\partial}{\partial x}[Kx\, p(x,t)] + \frac{D}{2}\frac{\partial^2}{\partial x^2}p(x,t)$$

where $D = \sigma^2$.
The Variance-Preserving scheme is to set $\frac{D}{2K} = 1$ and $K = 1$ in the Fokker Planck. The corresponding SDE becomes:

$$dx = -xdt + \sqrt{2}\, dW$$

which has solution

$$x_t = x_{t-1}e^{-\Delta t} + \mathbf{z}\sqrt{1 - e^{-2\Delta t}} \tag{2}$$

the stead state of which is $N(0, \frac{D}{2K} = 1)$, a standard gaussian.

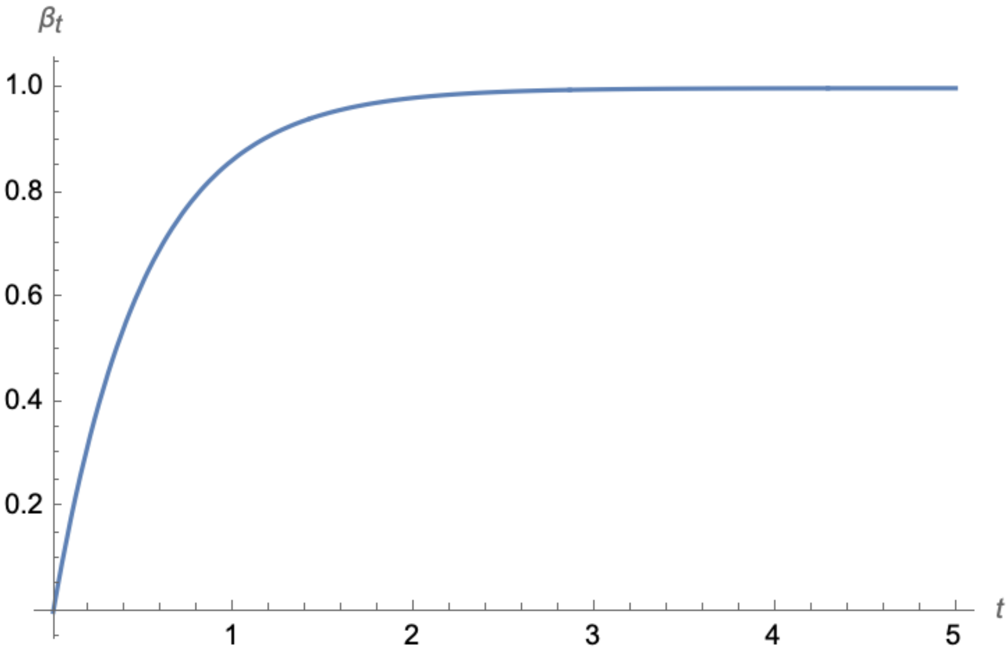
In DDPM, (2) is often discretized and re-parametrized as:

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

Comparison with (2) implies:

$$\beta_t = 1 - e^{-2\Delta t}$$

In other words, noise level $\beta_t$ determines time step $\Delta t$ in the corresponding continuous SDE.




Indeed, the longer the time step, the more noise is injected during that step.

## Connection to Hierarchical VAE

@yangDiffusionModelsComprehensive2022: Score SDE can be viewed as the continuous limit of hierarchical VAEs [4].

## References

---

1. @andersonReversetimeDiffusionEquation1982↩
2. @yangDiffusionModelsComprehensive2022 section 2.2 and @mengSDEditGuidedImage2022↩↩
3. Sigma is defined differently:
   $\sigma_t$ = stepwise reversal noise in @hoDenoisingDiffusionProbabilistic2020
   $\sigma_t = \sqrt{1-\bar{\alpha}_t}$, or the aggregate forward noise in @yangDiffusionModelsComprehensive2022↩
4. @vahdatNVAEDeepHierarchical2021↩