



Denoising Diffusion Probabilistic Models

Kshitij Ambilduke, Théo Bassérás, Nemanja Vujadinović

ENS Paris Saclay

Introduction

While traditional methods like VAEs and GANs have seen success, score-based generative modeling has emerged as a robust paradigm that avoids direct likelihood estimation by learning the score function ($\nabla_x \log p(x)$). A key insight that links this theory to practice is that training a denoising autoencoder to reconstruct clean data from Gaussian noise implicitly estimates the score of the data distribution. Hence, the generative process in DDPMs can be viewed as a specific parameterisation of Annealed Langevin Dynamics. Besides this, DDPMs can also be expressed using several equivalent objectives [1].

Contributions

- Implemented a modular DDPM framework with theoretically equivalent objectives.
- Incorporated Learned Variance to challenge the fixed covariance assumption.
- Implemented Classifier-Free Guidance for conditional generation without classifiers.

Formulation of DDPM

Forward process (Encoding)

Unlike VAEs, instead of having a learnt forward process, in DDPMs we have a fixed process defined by a Markov chain $q(x_t | x_{t-1}) = \mathcal{N}(x_t; \sqrt{\alpha_t}x_{t-1}, (1 - \alpha_t)I)$. Using this, and the bayes rule, we can show that $q(x_{t-1} | x_t, x_0) = \mathcal{N}(x_{t-1}; \mu_q(x_t, x_0), \Sigma_q)$ where,

$$\mu_q(x_t, x_0) = \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})x_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)x_0}{1 - \bar{\alpha}_t}, \quad \Sigma_q = \frac{(1 - \alpha_t)(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} I, \quad \bar{\alpha} = \prod_{i=1}^T \alpha_i.$$

Reverse process (Decoding)

We define a parameterised Markov chain that iteratively removes noise to reconstruct the data distribution $p_\theta(x_{t-1} | x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t), \Sigma_\theta(x_t))$.

ELBO

$$J_\theta(q) \propto \mathbb{E}_{q(x_1|x_0)}[\log p_\theta(x_0 | x_1)] - \sum_{t=2}^T \mathbb{E}_{q(x_t|x_0)}[\mathbb{D}_{\text{KL}}(q(x_{t-1} | x_t, x_0) \| p_\theta(x_{t-1} | x_t))].$$

Since by construction we know the distributions, we can write the loss function as:

$$L_{\mu_\theta} = \mathbb{E}_{q(x_1|x_0)} \left[\frac{1}{2\sigma_q^2} \|x_0 - \mu_\theta(x_1)\|_2^2 \right] + \mathbb{E}_{q(x_t|x_0), t \sim \mathcal{U}(2, T)} \left[\frac{1}{2\sigma_q^2} \|\mu_\theta(x_t, t) - \mu_q(x_t, x_0)\|_2^2 \right]$$

Unified loss framework

Clean image prediction

From the formulation of DDPM, we can see that $\mu_q(x_t, x_0) = Ax_t + Bx_0$, where A and B are scalars. Since this corresponds to scaling, we can reparametrize our new objective. Using this reparameterization, we can combine the two separate loss terms into a single one.

$$L_{x_0} = \mathbb{E}_{q(x_t|x_0), t \sim \mathcal{U}(1, t)} \left[\frac{1}{2\sigma_q^2} \frac{(1 - \alpha_t)^2}{(1 - \bar{\alpha}_t)^2} \bar{\alpha}_{t-1} \|\hat{x}_\theta(x_t) - x_0\|_2^2 \right]$$

Noise prediction

Alternatively, we can express x_0 in terms of the noise ϵ_t added during the forward process. In particular $x_0 = (x_t - \sqrt{1 - \bar{\alpha}_t}\epsilon_t)/\sqrt{\bar{\alpha}_t}$ which allows us to express the loss function in terms of predicting the added noise.

$$L_\epsilon = \mathbb{E}_{q(x_t|x_0), t \sim \mathcal{U}(1, t)} \left[\frac{1}{2\sigma_q^2} \frac{(1 - \alpha_t)^2}{(1 - \bar{\alpha}_t)\alpha_t} \|\epsilon_t - \hat{\epsilon}_\theta(x_t)\|_2^2 \right]$$

Score prediction

We can interpret DDPMs as score predictors by utilizing Tweedie's Formula. For a variable $x_t \sim \mathcal{N}(\mu_{x_t}, \Sigma_{x_t})$, the best estimate of the mean given the observation is:

$$\mathbb{E}[\mu_{x_t} | x_t] = x_t + \Sigma_{x_t} \nabla_{x_t} \log p(x_t)$$

Using this estimate for the distribution $x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}x_t + \frac{1 - \bar{\alpha}_t}{\sqrt{\bar{\alpha}_t}}\nabla_{x_t} \log p(x_t)$ and rearranging the equation, we get $q(x_t | x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1 - \bar{\alpha}_t)I)$ and hence, the corresponding loss becomes:

$$L_{s_\theta} = \mathbb{E}_{q(x_t|x_0), t \sim \mathcal{U}(1, t)} \left[\frac{1}{2\sigma_q^2} \frac{(1 - \alpha_t)^2}{\alpha_t} \|\nabla_{x_t} \log p(x_t) - s_\theta(x_t)\|_2^2 \right]$$

Simplified losses

- Following prior work [2], we disregard the constants and use only the norm difference for optimizing L_ϵ :

$$L_\epsilon = \|\epsilon_t - \hat{\epsilon}_\theta(x_t)\|_2^2$$
- Next, we substitute $\epsilon_t = -\nabla_{x_t} \log p(x_t | x_0) \sqrt{1 - \bar{\alpha}_t}$ into L_ϵ and optimize the following:

$$L_{s_\theta} = (1 - \bar{\alpha}_t) \|\nabla_{x_t} \log p(x_t | x_0) - s_\theta(x_t)\|_2^2$$
- Lastly, since the previous substitution yielded suboptimal results, we again disregard constants and optimize L_{x_0} with:

$$L_{x_0} = \|x_0 - \hat{x}_\theta(x_t)\|_2^2$$

Learned variance

Following previous findings that DDPMs exhibit weaker log-likelihood performance [3], we test a network that jointly learns both mean and the reverse process variance. We therefore extend it with an additional variance head v and obtain the variance as follows:

$$\Sigma_\theta(x_t, t) = \exp(v \log \beta_t + (1 - v) \log \tilde{\beta}_t).$$

Training objectives:

- Mean term: noise prediction loss L_ϵ
- Variance term: VLB loss L_{VLB} derived from ELBO

Final loss is then computed as follows:

$$L_{\text{learned}} = L_\epsilon + \lambda L_{\text{VLB}}, \quad \lambda = 1000.$$

During sampling, the learned variance replaces the fixed variance.

Classifier free guidance

Using the bayes rule, we can get the score for conditional generation as simply the sum of the unconditional score and the gradient from an external classifier.

$$\nabla \log \tilde{p}(x|y) = \nabla_x \log p(x) + \nabla_x \log p(y|x)$$

To avoid training a separate (and often unstable) classifier, we introduce a guidance weight w to scale the classifier term [4]. We then apply Bayes' rule again to rewrite the classifier term entirely using the generative model:

$$\nabla \log \tilde{p}(x|y) = (1 + w) \underbrace{\nabla \log p(x|y)}_{\text{Conditional}} - w \underbrace{\nabla \log p(x)}_{\text{Unconditional}}$$

Implementation note

We train a single model to predict both. During sampling, we pass the class label y to get the conditional score, and a null label \emptyset to get the unconditional score.

Sampling

Sampling is performed by reversing the diffusion process. Starting from pure noise, we iteratively sample from the reverse transition $p_\theta(x_{t-1}|x_t)$ along the chain $x_T \rightarrow \dots \rightarrow x_0$.

Update Rules

The mean of the reverse step depends on the chosen parameterisation of the loss function. The transition to x_{t-1} can be formulated in equivalent ways, where $z \sim \mathcal{N}(0, I)$ is the stochastic noise term:

$$\begin{aligned} [\mu_\theta] \quad x_{t-1} &= \mu_\theta(x_t, t) + \sigma_t z \\ [x_0] \quad x_{t-1} &= \left(\frac{(1 - \bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1 - \bar{\alpha}_t} \right) x_t + \left[\frac{(1 - \alpha_t)\sqrt{\bar{\alpha}_{t-1}}}{1 - \bar{\alpha}_t} \right] \hat{x}_\theta(x_t, t) + \sigma_t z \\ [\epsilon_\theta] \quad x_{t-1} &= \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right) + \sigma_t z \\ [s_\theta] \quad x_{t-1} &= \frac{1}{\sqrt{\alpha_t}} (x_t + (1 - \alpha_t)s_\theta(x_t, t)) + \sigma_t z \end{aligned}$$

It is worth noticing that the score based update step is a specific parameterization of Annealed Langevin Dynamics.

Results



Loss type	FID	FID Bin.	NLL
MNIST split topline	1.12	0.74	-
x_0 prediction	49.10	9.89	-
ϵ prediction	15.75	10.13	1.21
$s_\theta(x_t)$ prediction	78.83	17.36	-
$s_\theta(x_t y)$ prediction ($w = 5$)	74.21	25.16	-
$s_\theta(x_t y)$ prediction ($w = 1$)	61.91	15.69	-
Learned variance	58.89	14.45	1.1

Findings

- Uniform Weighting Enhances Quality:** Discarding variational lower bound terms in favor of uniform weighting significantly improves sample quality. This approach penalizes reconstruction errors equally across all time steps, whereas standard weighting fails to sufficiently penalize errors at high noise levels.
- Guidance Scale Trade-off:** While classifier guidance directs generation toward specific classes, a high guidance scale w forces the model to focus only on the most probable features. This reduces intra-class diversity and degrades FID scores, whereas a balanced scale ($w = 1$) optimizes both quality and diversity.
- Learned Variance Boosts Likelihood:** Incorporating learned variance enhances NLL scores without modifying the model architecture or degrading sample quality (FID).

References

- [1] Calvin Luo. *Understanding Diffusion Models: A Unified Perspective*, 2022.
- [2] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models, 2020.
- [3] A. Nichol and P. Dhariwal. Improved denoising diffusion probabilistic models, 2021.
- [4] Jonathan Ho and Tim Salimans. Classifier-Free Diffusion Guidance, 2022.