DDPM Diffusion Models from Scratch

Evan Dramko - Aug 12 2026

Contents

1 What Problem Is a Diffusion Model Trying to Solve?

Suppose we have a dataset (ex: images of dogs). As with all machine learning assume that all seen data is an independent sample from some unknown "oracle" data distribution (ex: the distributions of all possible images of dogs), $p^\star$, seen by:

$$x_0 \sim p^\star$$

We do not know an analytic formula for $p^\star$. We only have samples from it.

The goal of generative modeling is to construct a procedure that can produce *new* samples distributed approximately according to the same distribution:

$$\hat{x}_0 \sim p^\star$$

Directly learning how to sample from a complicated, high-dimensional distribution is difficult. Diffusion models replace this problem with a sequence of easier problems.

The high-level strategy is:

$$p^\star \longrightarrow p_{\Delta t} \longrightarrow p_{2\Delta t} \longrightarrow \cdots \longrightarrow p_T \approx q,$$

where q is a simple distribution from which we know how to sample, usually a Gaussian.

The forward direction deliberately destroys structure. The learned reverse direction reconstructs samples from the data distribution.

Core idea: Instead of learning one enormous transformation from random noise directly into data, construct many neighboring distributions and learn how to move backward between neighboring distributions.

2 Forward Diffusion: Destroying Data with Gaussian Noise

2.1 The basic random walk

Start with a real data sample x_0. The simplest Gaussian forward diffusion repeatedly adds independent zero-mean Gaussian noise:

$$x_{k+1} = x_k + \eta_k, \qquad \eta_k \sim \mathcal{N}(0, \sigma^2 I).$$

Thus,

$$x_1 = x_0 + \eta_0,$$

$$x_2 = x_0 + \eta_0 + \eta_1,$$

and after T steps,

$$x_T = x_0 + \sum_{k=0}^{T-1} \eta_k.$$

Because independent Gaussian random variables add to another Gaussian,

$$\sum_{k=0}^{T-1} \eta_k \sim \mathcal{N}(0, T\sigma^2 I)$$

Therefore,

$$x_T \mid x_0 \sim \mathcal{N}(x_0, T\sigma^2 I)$$

This is a random walk. Its expected displacement is zero:

$$\mathbb{E}[x_T - x_0] = 0,$$

but its variance grows with the number of steps:

$$\mathrm{Var}(x_T - x_0) = T\sigma^2 I$$

So "zero-mean noise" does *not* mean that the state stays close to x_0. It means that an ensemble of random walks remains centered around x_0, while becoming increasingly spread out.

2.2 Why scale the noise with $\sqrt{\Delta t}$?

We want to think of diffusion over a fixed time interval, say $t \in [0, 1]$, with

$$\Delta t = \frac{1}{T}$$

If we kept the variance of each step fixed while increasing T, then the terminal variance would grow without bound:

$$T\sigma^2 \to \infty$$

Instead, choose the per-step variance to be

$$\sigma_q^2 \Delta t$$

Equivalently, the standard deviation of each step is

$$\sigma_q \sqrt{\Delta t}$$

The forward update is therefore

$$\boxed{x_{t+\Delta t} = x_t + \eta_t, \qquad \eta_t \sim \mathcal{N}(0, \sigma_q^2 \Delta t\, I)}$$

After $T = 1/\Delta t$ steps, the accumulated noise variance is

$$T(\sigma_q^2 \Delta t) = \frac{1}{\Delta t} \sigma_q^2 \Delta t = \sigma_q^2$$

Thus the total noise scale is independent of how finely we discretize time.
More generally, after time t,

$$x_t \mid x_0 \sim \mathcal{N}(x_0, \sigma_q^2 t\, I)$$

This gives a particularly useful shortcut:

$$\boxed{x_t = x_0 + \sigma_q \sqrt{t}\, \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I)}$$

We therefore do *not* need to simulate every preceding forward step in order to construct a training example at time t.

3 What Does It Mean to Reverse Diffusion?

Suppose we are currently given a sample

$$x_t \sim p_t$$

A reverse sampler should produce a new sample whose *marginal distribution* is $p_{t-\Delta t}$.

The most direct possible reverse sampler is to sample from the true conditional distribution

$$\boxed{p(x_{t-\Delta t} \mid x_t)}$$

If we could do this exactly at every step, then

$$x_T \sim p_T$$

could be transformed recursively into

$$x_{T-\Delta t} \sim p_{T-\Delta t}, \quad x_{T-2\Delta t} \sim p_{T-2\Delta t}, \quad \ldots, \quad x_0 \sim p^\star$$

The problem is that $p(x_{t-\Delta t} \mid x_t)$ is initially unknown.

4 Why the Reverse Conditional Is Approximately Gaussian

The forward transition is known exactly:

$$p(x_t \mid x_{t-\Delta t}) = \mathcal{N}(x_t; x_{t-\Delta t}, \sigma_q^2 \Delta t\, I)$$

Bayes' rule gives

$$p(x_{t-\Delta t} \mid x_t) = \frac{p(x_t \mid x_{t-\Delta t}) p_{t-\Delta t}(x_{t-\Delta t})}{p_t(x_t)}$$

When studying this as a function of $x_{t-\Delta t}$, the observed x_t is fixed. Therefore $p_t(x_t)$ is merely a normalization factor.

Taking logs is useful because products become sums:

$$\log p(x_{t-\Delta t} \mid x_t) = \log p(x_t \mid x_{t-\Delta t}) + \log p_{t-\Delta t}(x_{t-\Delta t}) - \log p_t(x_t)$$

Terms depending only on the fixed x_t can be dropped when determining the *shape* of the conditional distribution.

For small Δt, approximate:

$$p_{t-\Delta t}(\cdot) = p_t(\cdot) + O(\Delta t),$$

and Taylor expands the log density around x_t. Since a typical diffusion displacement has size

$$x_{t-\Delta t} - x_t = O(\sqrt{\Delta t}),$$

the second-order spatial Taylor terms are $O(\Delta t)$.

Keeping the leading terms gives

$$\log p(x_{t-\Delta t} \mid x_t) \approx -\frac{\|x_{t-\Delta t} - x_t\|^2}{2\sigma_q^2 \Delta t} + \langle \nabla_x \log p_t(x_t), x_{t-\Delta t} - x_t \rangle + C$$

Completing the square yields

$$\log p(x_{t-\Delta t} \mid x_t) \approx -\frac{\left\| x_{t-\Delta t} - \left(x_t + \sigma_q^2 \Delta t \nabla_x \log p_t(x_t)\right) \right\|^2}{2\sigma_q^2 \Delta t} + C$$

This has the form of a Gaussian log density. Hence

$$\boxed{p(x_{t-\Delta t} \mid x_t) \approx \mathcal{N}\left(\mu_{t-\Delta t}(x_t), \sigma_q^2 \Delta t\, I\right),}$$

with

$$\boxed{\mu_{t-\Delta t}(x_t) = x_t + \sigma_q^2 \Delta t \nabla_x \log p_t(x_t)}$$

The quantity

$$\nabla_x \log p_t(x)$$

is called the *score*.

For sufficiently small steps, the forward process tells us the approximate *form* and variance of the reverse conditional. The principal unknown is its mean.

5 Why the Mean Must Be Learned

Define

$$\boxed{\mu_{t-\Delta t}(z) = \mathbb{E}[x_{t-\Delta t} \mid x_t = z]}$$

Notice that $\mu_{t-\Delta t}$ is a *function*. Different current states z generally imply different conditional means.

For example,

$$x_t = z_1 \quad \Longrightarrow \quad p(x_{t-\Delta t} \mid x_t = z_1) \approx \mathcal{N}(\mu_{t-\Delta t}(z_1), \sigma_q^2 \Delta t I),$$

while

$$x_t = z_2 \quad \Longrightarrow \quad p(x_{t-\Delta t} \mid x_t = z_2) \approx \mathcal{N}(\mu_{t-\Delta t}(z_2), \sigma_q^2 \Delta t I)$$

The variance is approximately known from the forward process. What changes with the current state is the mean.

This is the part parameterized by a neural network:

$$f_\theta(x_t, t) \approx \mu_{t-\Delta t}(x_t)$$

Time t is also supplied because the correct denoising behavior depends on the noise level. Some implementations directly supply the total expected noise, $\sigma_{total}(t) = \sigma_q \sqrt{t}$, rather than supplying t. The two are largely equivalent, as the only information in t is how much noise to expect. However, for mathematical clarity and connections to future extensions of diffusion based on differential equations, it is more common to see t denoted in derivations.

6 Why Ordinary MSE Regression Learns the Conditional Mean

A fundamental regression identity is

$$\boxed{\arg\min_f \mathbb{E}\left[\|f(X) - Y\|^2\right] = \mathbb{E}[Y \mid X]}$$

Apply this with

$$X = x_t, \qquad Y = x_{t-\Delta t}$$

Then

$$\arg\min_f \mathbb{E}\left[\|f(x_t) - x_{t-\Delta t}\|^2\right] = \mathbb{E}[x_{t-\Delta t} \mid x_t]$$

Therefore, we can learn the reverse mean with an ordinary supervised regression problem:

$$\boxed{\mathcal{L}(\theta) = \mathbb{E}\left[\|f_\theta(x_t, t) - x_{t-\Delta t}\|^2\right]}$$

This is one of the central simplifications of DDPM: learning a complicated generative distribution has been reduced to regression.

7 How Training Data Are Generated

The forward process is completely known, so it acts as a training-data generator.

A direct training procedure is:

1. Sample a real example
$$x_0 \sim p^\star$$
2. Sample a time
$$t \sim \text{Uniform}[0, 1]$$
3. Directly construct
$$x_t = x_0 + \mathcal{N}(0, \sigma_q^2 t\, I)$$
4. Add one more forward noise step:
$$x_{t+\Delta t} = x_t + \mathcal{N}(0, \sigma_q^2 \Delta t\, I)$$
5. Ask the network, f, to predict x_t from $x_{t+\Delta t}$:
$$\hat{x}_t = f_\theta(x_{t+\Delta t}, t + \Delta t)$$
6. Minimize
$$\mathcal{L} = \|\hat{x}_t - x_t\|^2$$

Crucially, standard DDPM training in this presentation does *not* require unrolling the entire reverse generation process and backpropagating through all reverse steps.

Each training example is a local denoising regression problem generated using the known forward process.

8 Why Zero-Mean Forward Noise Can Produce a Nonzero Reverse Mean

Each forward increment satisfies

$$\mathbb{E}[\eta_i] = 0$$

This does *not* imply

$$\mathbb{E}[\eta_i \mid x_t] = 0$$

The distinction is conditioning.

Suppose, for intuition, that

$$x_2 = \eta_1 + \eta_2, \qquad \eta_1, \eta_2 \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$

Before observing the endpoint,

$$\mathbb{E}[\eta_1] = \mathbb{E}[\eta_2] = 0$$

But if we condition on

$$x_2 = 4,$$

then we know

$$\eta_1 + \eta_2 = 4$$

By symmetry,

$$\mathbb{E}[\eta_1 \mid x_2 = 4] = \mathbb{E}[\eta_2 \mid x_2 = 4] = 2$$

Thus zero unconditional mean is perfectly compatible with a nonzero conditional mean.

This is exactly the type of inference required in reverse diffusion: given where the random walk currently is, infer what its earlier increments were likely to have been.

9 Variance Reduction: Predicting x_0

Directly predicting $x_{t-\Delta t}$ from x_t is possible, but the two states differ by only a tiny random step. We show that a lower-variance target can be obtained by predicting x_0 from each x_t, rather than predicting $x_{t-\Delta t}$.

Write the accumulated forward process as

$$x_t = x_0 + \eta_1 + \eta_2 + \cdots + \eta_N, \qquad N = \frac{t}{\Delta t}$$

The final forward increment obeys

$$x_t = x_{t-\Delta t} + \eta_N$$

Therefore,

$$x_{t-\Delta t} - x_t = -\eta_N$$

Conditioned on x_t, the individual Gaussian increments are symmetric. Consequently,

$$\mathbb{E}[\eta_1 \mid x_t] = \mathbb{E}[\eta_2 \mid x_t] = \cdots = \mathbb{E}[\eta_N \mid x_t]$$

Their sum is

$$\sum_{i=1}^{N} \eta_i = x_t - x_0$$

Hence the expected contribution from one increment is the expected total contribution divided by the number of increments:

$$\mathbb{E}[\eta_N \mid x_t] = \frac{1}{N}\mathbb{E}[x_t - x_0 \mid x_t]$$

Since

$$N = \frac{t}{\Delta t},$$

we obtain

$$\boxed{\mathbb{E}[x_{t-\Delta t} - x_t \mid x_t] = \frac{\Delta t}{t}\mathbb{E}[x_0 - x_t \mid x_t]}$$

Equivalently,

$$\boxed{\mathbb{E}[x_{t-\Delta t} \mid x_t] = \frac{\Delta t}{t}\mathbb{E}[x_0 \mid x_t] + \left(1 - \frac{\Delta t}{t}\right) x_t}$$

Thus, instead of directly learning the tiny one-step correction, a model may learn

$$f_\theta(x_t, t) \approx \mathbb{E}[x_0 \mid x_t],$$

and convert this prediction into the required one-step reverse mean.

A critical point is that "predicting x_0" means predicting the *conditional expectation*

$$\mathbb{E}[x_0 \mid x_t],$$

not sampling a particular clean image from $p(x_0 \mid x_t)$. At high noise levels this conditional expectation can itself look like an average or blurry mixture of multiple plausible clean examples.

10 The DDPM Generation Process

Once the reverse mean estimator has been trained, generation begins from the simple terminal distribution:

$$x_T \sim q$$

In Δt normalized notation, $T = 1$ and

$$x_1 \sim \mathcal{N}(0, \sigma_q^2 I)$$

At each reverse step, compute the estimated reverse mean and add a newly sampled Gaussian perturbation:

$$\boxed{x_{t-\Delta t} = f_\theta(x_t, t) + \tilde{\eta}_t, \qquad \tilde{\eta}_t \sim \mathcal{N}(0, \sigma_q^2 \Delta t\, I)}$$

Then repeat:

$$x_T \to x_{T-\Delta t} \to x_{T-2\Delta t} \to \cdots \to x_0$$

The reverse noise $\tilde{\eta}_t$ is *fresh noise*. It is not the original forward noise being recovered and subtracted.

11 Why Add Noise While Denoising?

At first this seems contradictory. Forward diffusion destroys an image by adding Gaussian noise, yet DDPM generation also adds Gaussian noise at every reverse step.

The resolution is that the reverse process is not trying to reconstruct a unique previous state.

Given x_t, there are generally many possible $x_{t-\Delta t}$ values that could have produced it. The correct reverse object is therefore a distribution:

$$p(x_{t-\Delta t} \mid x_t)$$

For small Δt,

$$p(x_{t-\Delta t} \mid x_t) \approx \mathcal{N}(\mu_{t-\Delta t}(x_t), \sigma_q^2 \Delta t I)$$

If we used only

$$x_{t-\Delta t} = \mu_{t-\Delta t}(x_t),$$

we would always select the conditional mean rather than sample from the reverse conditional.

Instead,

$$x_{t-\Delta t} = \underbrace{\mu_{t-\Delta t}(x_t)}_{\text{directed reverse movement}} + \underbrace{\tilde{\eta}_t}_{\text{stochastic spread}}$$

The forward process is

$$x_{t+\Delta t} = \underbrace{x_t}_{\text{unchanged mean}} + \underbrace{\eta_t}_{\text{random spread}},$$

whereas the reverse process has a shifted mean:

$$x_{t-\Delta t} = \underbrace{x_t + \sigma_q^2 \Delta t \nabla \log p_t(x_t)}_{\text{shift toward higher probability}} + \underbrace{\tilde{\eta}_t}_{\text{random spread}}$$

The two Gaussian noises should not be imagined as "canceling." The reverse process instead has a systematic drift in its mean that causes the *distribution of samples* to evolve back toward the data distribution even while individual reverse trajectories remain stochastic.

> DDPM reverses the evolution of probability distributions. It does not retrace an individual forward random walk.

11.1 Further Details on Addition of Noise in Reverse Direction

At first, the reverse diffusion update may seem counterintuitive. The forward process destroys structure by repeatedly adding Gaussian noise, yet the reverse process also adds Gaussian noise:

$$x_{t-\Delta t} = \mu_{t-\Delta t}(x_t) + \sigma_q \sqrt{\Delta t}\, \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I)$$

Why not simply use

$$x_{t-\Delta t} = \mu_{t-\Delta t}(x_t)$$

and move deterministically toward a clean image?

The key is that the reverse mean and the reverse sample serve two different purposes. Recall that

$$\mu_{t-\Delta t}(x_t) = \mathbb{E}[x_{t-\Delta t} \mid x_t]$$

This is the *average* of the possible previous states given the current noisy state. It is not, in general, itself a sample from the reverse conditional distribution.

For sufficiently small diffusion steps, the reverse conditional is approximately

$$p(x_{t-\Delta t} \mid x_t) \approx \mathcal{N}\left(\mu_{t-\Delta t}(x_t), \sigma_q^2 \Delta t\, I\right)$$

Therefore, if we want to draw a sample from this distribution, we need both its mean and its variance. Sampling from the Gaussian gives

$$x_{t-\Delta t} = \underbrace{\mu_{t-\Delta t}(x_t)}_{\text{conditional center}} + \underbrace{\sigma_q \sqrt{\Delta t}\, \epsilon}_{\text{conditional spread}}$$

The added noise should therefore not be interpreted as making our estimate of the mean less accurate. The mean has already been estimated. Instead, the noise converts that mean into a random sample from the full reverse conditional distribution.

A Simple Numerical Example

Suppose, for illustration, that for a particular current state x_t, the true reverse conditional is

$$x_{t-\Delta t} \mid x_t \sim \mathcal{N}(5, 1).$$

Its conditional mean is

$$\mathbb{E}[x_{t-\Delta t} \mid x_t] = 5$$

If we perform the reverse step deterministically,

$$x_{t-\Delta t} = 5,$$

then every reverse trajectory passing through this particular x_t is assigned exactly the same previous state.

However, the actual conditional distribution says that plausible previous states should be distributed around 5. For example, we might obtain values such as

$$4.3, \qquad 5.1, \qquad 5.8, \qquad 3.9,$$

with values closer to 5 being more probable.

To reproduce this distribution, we instead sample

$$x_{t-\Delta t} = 5 + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, 1)$$

A sampled value such as 5.8 is not a worse estimate of the mean than 5. It is not intended to estimate the mean at all. Rather, 5.8 is one possible sample from a distribution whose mean is 5.

Why Using Only the Mean Loses Information

The distinction can also be seen directly through the variance. Conditioned on x_t, the desired reverse distribution has

$$\mathrm{Var}(x_{t-\Delta t} \mid x_t) = \sigma_q^2 \Delta t \, I$$

If we instead define

$$x_{t-\Delta t} = \mu_{t-\Delta t}(x_t),$$

then, once x_t is fixed, the result is deterministic. Therefore,

$$\mathrm{Var}(x_{t-\Delta t} \mid x_t) = 0$$

We have collapsed an entire distribution of plausible previous states onto a single conditional average.

Adding the Gaussian term restores the required variance:

$$\begin{aligned} \mathrm{Var}\left[\mu_{t-\Delta t}(x_t) + \sigma_q \sqrt{\Delta t}\, \epsilon \mid x_t\right] &= \mathrm{Var}\left[\sigma_q \sqrt{\Delta t}\, \epsilon\right] \\ &= \sigma_q^2 \Delta t \, I \end{aligned}$$

Thus, the two pieces of the reverse update have complementary roles:

$\mu_{t-\Delta t}(x_t)$	provides the correct conditional center,
$\sigma_q \sqrt{\Delta t}\, \epsilon$	provides the correct conditional spread.

Connection to "Blurry" Conditional Means

This also provides intuition for why repeatedly selecting only a conditional mean can be undesirable. Suppose a noisy state is consistent with several different plausible previous states. The conditional mean averages over these possibilities:

$$\mu_{t-\Delta t}(x_t) = \mathbb{E}[x_{t-\Delta t} \mid x_t]$$

For image data, an average over multiple plausible structures can produce a state that resembles none of the individual possibilities. This is the same reason that image models trained only to minimize pixel-wise mean squared error can produce blurry predictions when several distinct outputs are plausible.

Sampling from the reverse conditional instead allows different trajectories to select different plausible previous states rather than forcing every trajectory through their conditional average.

It is important, however, not to overinterpret this picture. At an intermediate diffusion time, a sampled $x_{t-\Delta t}$ is not necessarily a clean or realistic image. It is instead a plausible sample from the appropriate *slightly less noisy* distribution $p_{t-\Delta t}$. Repeating these stochastic reverse transitions eventually produces

$$p_T \longrightarrow p_{T-\Delta t} \longrightarrow \cdots \longrightarrow p_0 = p^\star$$

> The Gaussian noise in a DDPM reverse step is not an error added to the predicted mean. The mean specifies where the reverse conditional is centered, while the noise supplies its required variance. Together, they produce a sample from the reverse conditional rather than merely its average.

12 Why the Small-Step Approximation Is Reasonable

The reverse conditional is only approximately Gaussian for a finite step size.

We quanitfy the per-step approximation using KL divergence:

$$D_{\mathrm{KL}}(P\|Q) = \mathbb{E}_{x\sim P}\left[\log \frac{P(x)}{Q(x)}\right]$$

A KL divergence of zero means the two probability distributions agree (up to sets of measure zero).

For a forward step with variance σ^2, the we have a bound of the form

$$D_{\mathrm{KL}}\left(\mathcal{N}(\mu_z, \sigma^2) \,\middle\|\, p(x_0 \mid x_1 = z)\right) \leq O(\sigma^4)$$

In the diffusion notation,

$$\sigma^2 = \sigma_q^2 \Delta t,$$

so

$$\sigma^4 = O(\Delta t^2)$$

There are

$$O(1/\Delta t)$$

steps over a fixed time interval. Roughly, summing the per-step errors gives

$$O(1/\Delta t)\, O(\Delta t^2) = O(\Delta t),$$

which vanishes as

$$\Delta t \to 0$$

Thus making the discretization finer simultaneously increases the number of steps and improves the quality of each local Gaussian approximation, with the latter improving sufficiently quickly.

13 Continuous Time and Stochastic Differential Equations

13.1 Refresher: Euler's method

For an ordinary differential equation

$$\frac{dx}{dt} = f(x,t),$$

Euler's method uses

$$x_{t+\Delta t} \approx x_t + f(x_t,t)\Delta t$$

As $\Delta t \to 0$, the discrete updates approach a continuous trajectory.

The explicit t appears because the derivative may depend both on the current state and on time. If it does not depend explicitly on time, the ODE can instead be written

$$\frac{dx}{dt} = f(x)$$

13.2 The stochastic version

Our discrete forward diffusion is

$$x_{t+\Delta t} = x_t + \sigma_q\sqrt{\Delta t}\,\xi, \qquad \xi \sim \mathcal{N}(0,I)$$

As $\Delta t \to 0$, this becomes the stochastic differential equation

$$\boxed{dx = \sigma_q\, dw,}$$

where w is Brownian motion.

The general SDE form is

$$\boxed{dx = f(x,t)\,dt + g(t)\,dw}$$

The first term is the deterministic *drift*; the second is stochastic diffusion.

A useful discrete interpretation is

$$x_{t+\Delta t} \approx x_t + f(x_t,t)\Delta t + g(t)\sqrt{\Delta t}\,\xi$$

This is the stochastic analogue of an Euler step.

13.3 Why Brownian motion is not an ordinary ODE

For an ordinary Euler update,

$$\Delta x = O(\Delta t).$$

For Brownian motion,

$$\Delta x = O(\sqrt{\Delta t})$$

Attempting to form an ordinary derivative gives

$$\frac{\Delta x}{\Delta t} = O\left(\frac{1}{\sqrt{\Delta t}}\right),$$

which does not approach a finite derivative as $\Delta t \to 0$.

Thus Brownian paths are not differentiable in the ordinary sense, motivating stochastic calculus and SDE notation.

13.4 The reverse-time SDE

For a general forward SDE

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

the reverse-time SDE is

$$\boxed{dx = \big(f(x,t) - g(t)^2 \nabla_x \log p_t(x)\big)\,dt + g(t)\,dw,}$$

when time is run backward.

For the simple forward diffusion

$$dx = \sigma_q\,dw,$$

we have

$$f = 0, \qquad g = \sigma_q,$$

so the reverse equation is

$$dx = -\sigma_q^2 \nabla_x \log p_t(x)\,dt + \sigma_q\,dw$$

The score appears again. This is the continuous-time counterpart of the mean shift derived earlier with Bayes' rule.

Discretizing this reverse SDE recovers the same DDPM-style update:

$$x_{t-\Delta t} \approx \mathbb{E}[x_{t-\Delta t} \mid x_t] + \mathcal{N}(0, \sigma_q^2 \Delta t I)$$

Thus the SDE viewpoint and the discrete DDPM derivation are two descriptions of the same basic reverse process in this simplified setting.

14 Training and Generation Side by Side

It is useful to separate what happens during *training* from what happens during *generation*.

Training

We possess real data:

$$x_0 \sim p^\star$$

We deliberately corrupt it:

$$x_0 \longrightarrow x_t \longrightarrow x_{t+\Delta t}$$

Because we generated the corruption ourselves, we know the regression target. We train

$$f_\theta(x_{t+\Delta t}, t + \Delta t)$$

using an MSE loss so that it estimates the appropriate conditional mean.

There is no need to start at x_T, run the complete reverse chain, compare the final result with x_0, and backpropagate through every generation step.

Generation

Now no clean image is provided.

We sample

$$x_T \sim q$$

and recursively apply the learned reverse transition:

$$x_{t-\Delta t} = f_\theta(x_t, t) + \mathcal{N}(0, \sigma_q^2 \Delta t I)$$

After enough reverse steps,

$$x_0$$

is approximately distributed according to the data distribution.

Training:	$x_0 \sim p^\star$	$\xrightarrow{\text{known noising}}$	x_t	$\xrightarrow{\text{supervised regression}}$ learn reverse information
Generation:	$x_T \sim q$	$\xrightarrow{\text{learned stochastic reversal}}$	$\cdots$	$\to x_0 \sim p^\star$

15 A Complete Mental Model of DDPM

The entire argument can now be summarized as follows.

1. **We have data but not its density.**

   We can sample

   $$x_0 \sim p^\star,$$

   but cannot directly sample new points from an analytic model of $p^\star$.

2. **Construct a known forward process.**

   Add small independent Gaussian increments:

   $$x_{t+\Delta t} = x_t + \mathcal{N}(0, \sigma_q^2 \Delta t I)$$

3. **The distribution gradually becomes simple.**

   After enough noising,

   $$p_T \approx q,$$

   where q is approximately Gaussian and easy to sample.

4. **Reverse generation can be done one small step at a time.**

   Ideally we would sample

   $$x_{t-\Delta t} \sim p(x_{t-\Delta t} \mid x_t)$$

5. **For sufficiently small steps, this reverse conditional is approximately Gaussian.**

   $$p(x_{t-\Delta t} \mid x_t) \approx \mathcal{N}(\mu_{t-\Delta t}(x_t), \sigma_q^2 \Delta t I)$$

6. **The difficult unknown is the mean.**

$$\mu_{t-\Delta t}(x_t) = \mathbb{E}[x_{t-\Delta t} \mid x_t]$$

7. **MSE regression learns conditional expectations.**

   Therefore a neural network can learn this mean from noisy examples generated by the known forward process.

8. **Practical parameterizations can instead predict x_0.**

   The expected clean point

   $$\mathbb{E}[x_0 \mid x_t]$$

   can be converted into the required one-step reverse mean.

9. **Generation begins with random noise.**

$$x_T \sim q$$

10. **Each reverse step samples from the learned Gaussian conditional.**

$$x_{t-\Delta t} = \mu_{t-\Delta t}(x_t) + \mathcal{N}(0, \sigma_q^2 \Delta t I)$$

11. **Repeated reverse sampling transports the distribution back toward the data distribution.**

$$q \to p_{T-\Delta t} \to \cdots \to p^\star$$

16 Common Conceptual Traps

"The forward noise has mean zero, so shouldn't the reverse mean also be zero?"

No. Forward increments have

$$\mathbb{E}[\eta] = 0,$$

but reverse inference uses conditional expectations such as

$$\mathbb{E}[\eta \mid x_t],$$

which need not be zero after observing where the random walk ended.

Toy Example: Why Zero-Mean Forward Noise Does Not Imply a Zero Reverse Mean

Consider a simple two-step random walk starting from a known point

$$x_0 = 0$$

We add two independent Gaussian noise increments,

$$\eta_1, \eta_2 \sim \mathcal{N}(0, 1),$$

so that

$$x_1 = x_0 + \eta_1, \qquad x_2 = x_1 + \eta_2 = 0 + \eta_1 + \eta_2$$

Before observing the outcome of the random walk, both noise increments have mean zero:

$$\mathbb{E}[\eta_1] = \mathbb{E}[\eta_2] = 0.$$

Consequently,

$$\mathbb{E}[x_2] = \mathbb{E}[\eta_1 + \eta_2] = 0$$

Thus, the *forward* random walk remains centered at zero, even though its variance increases.

Now suppose that we observe one particular endpoint:

$$x_2 = 4.$$

This observation gives us additional information. Since

$$x_2 = \eta_1 + \eta_2,$$

we now know that

$$\eta_1 + \eta_2 = 4$$

The two noise increments are statistically symmetric, so conditioned on their sum being 4, neither increment has any reason to account for more of the displacement than the other. Therefore,

$$\mathbb{E}[\eta_1 \mid x_2 = 4] = \mathbb{E}[\eta_2 \mid x_2 = 4] = 2$$

This does not contradict the fact that

$$\mathbb{E}[\eta_i] = 0$$

The two expectations answer different questions:

$$\boxed{\mathbb{E}[\eta_i] = 0 \qquad \text{but} \qquad \mathbb{E}[\eta_i \mid x_2 = 4] = 2}$$

The first averages over *all possible random walks*. The second considers only random walks that happened to end at $x_2 = 4$.

This explains why the reverse process can have a nonzero mean even though every forward noise increment has mean zero. In diffusion, once we observe the current noisy state x_t, that state contains information about the noise increments that must have occurred along the way. Therefore, in general,

$$\mathbb{E}[\eta_i \mid x_t] \neq 0$$

For this toy example, consider reversing the final step. Since

$$x_2 = x_1 + \eta_2,$$

we have

$$x_1 = x_2 - \eta_2$$

Conditioning on the observed value $x_2 = 4$,

$$\mathbb{E}[x_1 \mid x_2 = 4] = 4 - \mathbb{E}[\eta_2 \mid x_2 = 4] = 4 - 2 = 2$$

Thus the reverse conditional is centered at 2, not at 4 and not at 0:

$$\boxed{\mathbb{E}[x_1 \mid x_2 = 4] = 2}$$

The forward process has no directional drift because its Gaussian increments have unconditional mean zero. The reverse process, however, conditions on the observed noisy state. This conditioning creates a nonzero expected correction that moves the state back toward regions from which it was more likely to have originated.

"Does reverse diffusion subtract the original forward noise?"

No. During generation there is no original forward trajectory to retrace. DDPM samples a new plausible previous state from

$$p(x_{t-\Delta t} \mid x_t)$$

"Why add noise during denoising?"

Because the reverse conditional has nonzero variance. The shifted mean provides the systematic reverse movement, while the Gaussian term samples among plausible previous states.

"Does the network learn a Gaussian distribution?"

In this simplified presentation, the Gaussian form and variance come from the diffusion construction. The network learns the conditional mean (or an equivalent quantity from which that mean can be computed).

"Do we backpropagate through the complete generation trajectory?"

Not in the DDPM training procedure presented here. Training samples a real x_0, constructs noisy states using the known forward process, and applies an ordinary regression loss at a sampled time.

"Is the reverse process an inverse function?"

Not generally. Forward diffusion is stochastic and destroys information, so there is no unique inverse state. The natural inverse object is a conditional probability distribution.

17 Notation Cheat Sheet

Symbol	Meaning
$p^\star$	Unknown target/data distribution
x_0	Clean data sample, $x_0 \sim p^\star$
x_t	State after diffusion for time t
p_t	Marginal distribution of x_t
q	Easy-to-sample terminal/base distribution
Δt	Discrete timestep size
σ_q^2	Desired terminal noise variance in the simplified process
η_t	Forward Gaussian noise increment
$\mu_{t-\Delta t}(x_t)$	Mean of $p(x_{t-\Delta t} \mid x_t)$
f_θ	Neural network used to estimate reverse information
$\nabla_x \log p_t(x)$	Score of the noisy marginal distribution
dw	Infinitesimal Brownian-motion increment
$f(x,t)$	Drift term of a general SDE
$g(t)$	Diffusion/noise coefficient of a general SDE

18 Final Perspective

DDPM can be understood as a carefully constructed stochastic reversal problem.

We begin with a complicated distribution $p^\star$ that we can sample from only through a dataset. We define a forward random walk that gradually converts this distribution into an easy Gaussian-like distribution. Because each forward step is very small, its reverse conditional becomes approximately Gaussian as well. The forward construction tells us the reverse variance; the remaining unknown is the state-dependent conditional mean.

That mean is learnable through ordinary regression because MSE regression estimates conditional expectations. Training therefore uses the known forward process to synthesize noisy supervised examples. Generation then starts from Gaussian noise and repeatedly samples from the learned reverse conditional.

The essential chain of reasoning is

known Gaussian forward process
$\Downarrow$
simple local reverse conditionals
$\Downarrow$
learn their conditional means by regression
$\Downarrow$
start from q and repeatedly reverse
$\Downarrow$
obtain samples approximately from $p^\star$.

The SDE viewpoint is the continuous-time version of the same idea: as $\Delta t \to 0$, the discrete Gaussian random walk becomes Brownian motion, and the learned score supplies the drift required to reverse the diffusion.