Lecture 02: Score Based and Diffusion Models

Hanseul Kim

January 2025

1 Introduction

In this report, we look into motivation and mathematical proofs to score-based models and diffusion models.

2 Score-based generative model (recap)

2.1 Explicit to implicit score matching model

Score matching [3] was originally designed to learn non-normalized statistical models with i.i.d samples from an unknown data distribution. [9]
We use score network $s_\theta(x)$ to estimate $\nabla \log p_{\text{data}}(x)$. Then the objective function becomes,

$$J_{\text{ESP}} = \frac{1}{2}\mathbb{E}_{p_{\text{data}}(\mathbf{x})}[||tr(\nabla_{\mathbf{x}} s_\theta(\mathbf{x})) - \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})||_2^2]$$

$$= \frac{1}{2}\mathbb{E}_{p_{\text{data}}(\mathbf{x})}[s_\theta(\mathbf{x})^T s_\theta(\mathbf{x}) - 2 s_\theta(\mathbf{x})^T \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x}) + \nabla \log p_{\text{data}}(\mathbf{x})^T \nabla \log p_{\text{data}}(\mathbf{x})]$$

$$= \mathbb{E}_{p_{\text{data}}(\mathbf{x})}[tr(\nabla_{\mathbf{x}} s_\theta(\mathbf{x})) + \frac{1}{2}||s_\theta(\mathbf{x})||_2^2] + C$$

Limitation

- Computational time of trace n samples and d dimension takes computational complexity of $O(nd)$
- Prone to over-fitting due to data being finite and discrete

2.2 Denoising score matching model

Adding noise to the data helps to mediate over-fitting due to perturbed data creating more diverse region than original data manifold as seen on figure 1, 2

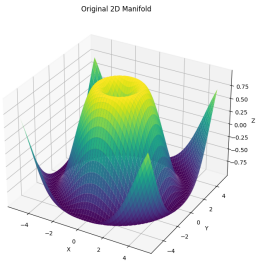

Figure 1: Original manifold

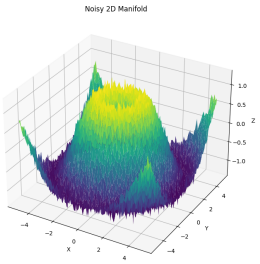

Figure 2: Gaussian perturbed manifold

$$J_{\mathrm{ESM}_{p_\sigma}}(\theta) = \frac{1}{2}\mathbb{E}_{p_\theta(\tilde{\mathbf{x}})}[||\nabla_{\tilde{\mathbf{x}}} \log p_\sigma(\tilde{\mathbf{x}}) - s_\theta(\tilde{\mathbf{x}}, \sigma)||^2]$$

$$\Rightarrow \frac{1}{2}\mathbb{E}_{p_\theta(\tilde{\mathbf{x}})}[||s_\theta(\tilde{\mathbf{x}}, \sigma)||^2] - \mathbb{E}_{p_\theta(\tilde{\mathbf{x}})}[s_\theta(\tilde{\mathbf{x}}, \sigma)^T \nabla_{\tilde{\mathbf{x}}} \log p_\sigma(\tilde{\mathbf{x}})]$$

$$\Rightarrow \frac{1}{2}\mathbb{E}_{p_\theta(\tilde{\mathbf{x}})}[||s_\theta(\tilde{\mathbf{x}}, \sigma)||^2] - \mathbb{E}_{p_\theta(\tilde{\mathbf{x}}, \mathbf{x})}[s_\theta(\tilde{\mathbf{x}}, \sigma)^T \nabla_{\tilde{\mathbf{x}}} \log p_\sigma(\tilde{\mathbf{x}}|\mathbf{x})]$$

$$= J_{\mathrm{DSM}_{p_\sigma}}(\theta) - C$$

$$J_{\mathrm{DSM}_{p_\sigma}}(\theta) = \frac{1}{2}\mathbb{E}_{q_\sigma(\tilde{\mathbf{x}}|\mathbf{x})p_{\mathrm{data}}(\mathbf{x})}[||s_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}}|\mathbf{x})||_2^2]$$

For small enough noise, $q_\sigma(\mathbf{x}) \approx p_{\mathrm{data}}(\mathbf{x})$ and $s_{\theta^*}(\mathbf{x}) = \nabla_{\mathbf{x}} \log q_\sigma(\mathbf{x}) \approx \nabla_{\mathbf{x}} \log p_{\mathrm{data}}(\mathbf{x})$

2.2.1 Sampling

Given initial sample of $x_0 \sim \pi(\mathbf{x})$ (π being prior distribution), new images can be sampled by $\epsilon \to 0, T \to \infty$

$$\tilde{\mathbf{x}}_t = \tilde{\mathbf{x}}_{t-1} + \frac{\epsilon}{2}\nabla_{\mathbf{x}} \log p(\tilde{\mathbf{x}}_{t-1}) + \sqrt{\epsilon}\mathbf{z}_t$$

We can sample using approximation function $s_\theta(\mathbf{x}) \approx \nabla_{\mathbf{x}} \log p_{\mathrm{data}}(\mathbf{x})$

2.3 Limitation

- The manifold hypothesis [1]

The data generated in the world are highly sparse and lies on low dimensional manifold. Related to the world being highly biased and might be related to curse of dimensionality [5] on figure 3

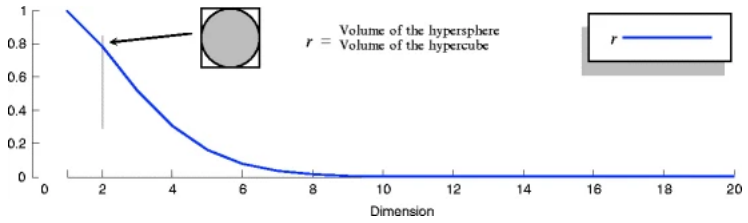

Figure 3: Curse of dimensionality

This distribution of dataset leads to broad *ambient space* [9], which learned gradient being inconsistent. This was shown in a toy experiment [9] seen on figure 4

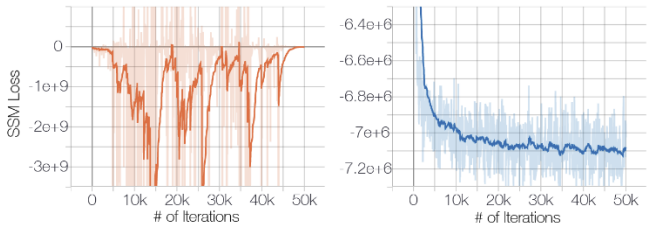

Figure 4: **Left**: Sliced score matching (SSM) loss w.r.t. iterations. No noise is added to data. **Right**: Same but data are perturbed with Gaussian noise

$$\text{SSL} = \mathbb{E}_{p_v}\mathbb{E}_{p_{\text{data}}}[\mathbf{v}^T\nabla_{\mathbf{x}}s_\theta\mathbf{v} + \frac{1}{2}||s_\theta(\mathbf{x})||_2^2]$$

$$\mathbb{E}_{p_{\text{data}}}[tr(\nabla_{\mathbf{x}}s_\theta(\mathbf{x}))] = \mathbb{E}_{p_{\text{data}}}[tr(\nabla_{\mathbf{x}}s_\theta(\mathbf{x})I)]$$

$$\mathbb{E}_{p_{\text{data}}}[tr(\nabla_{\mathbf{x}}s_\theta(\mathbf{x})\mathbb{E}_{\mathbf{v}}[\mathbf{v}\mathbf{v}^T])] = \mathbb{E}_{p_v}\mathbb{E}_{p_{\text{data}}}[tr(\nabla_{\mathbf{x}}s_\theta(\mathbf{x})\mathbf{v}\mathbf{v}^T)]$$

$$= \mathbb{E}_{p_v}\mathbb{E}_{p_{\text{data}}}[tr(\mathbf{v}^T\nabla_{\mathbf{x}}s_\theta(\mathbf{x})\mathbf{v})] = \mathbb{E}_{p_{\text{data}}}[\mathbf{v}^T\nabla_{\mathbf{x}}s_\theta(\mathbf{x})\mathbf{v}]$$

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

This gradient error in *ambient space* leads to error when generating new data. As seen on figure 5 [9] when $\mathbf{x}_t$ is at *ambient space*, it will have harder time getting out of such space using decreasing ϵ. And for the center region, the $d\mathbf{x}$ gradient will be just random direction which makes it even harder.

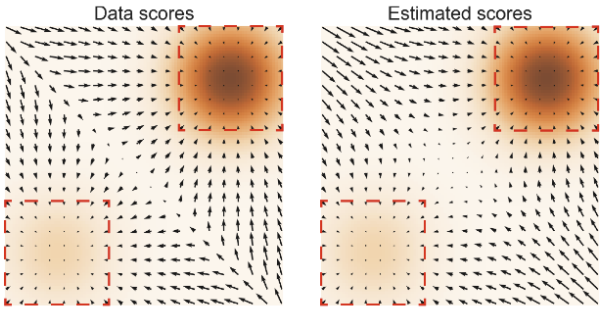

Figure 5: **Left:** $\nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})$; **Right**: $s_\theta(\mathbf{x})$
. darker color implies higher density.

- Slow mixing of Langevin dynamics

When there are two distinct modes of the data distribution separated by low density reason, score models will not learn different weights of mode distribution. Let data distribution consists of two distinct distribution.

$$p_{\text{data}}(\mathbf{x}) = \pi p_1(\mathbf{x}) + (1-\pi)p_2(\mathbf{x})$$

$$\begin{aligned}\nabla_{\mathbf{x}}(\log p_{\text{data}}) &= \nabla_{\mathbf{x}}(\log \pi + \log(1-\pi) + \log p_1(\mathbf{x}) + \log p_2(\mathbf{x})) \\ &= \nabla_{\mathbf{x}}(\log p_1(\mathbf{x}) + \log p_2(\mathbf{x}))\end{aligned}$$

Which does not depends on π. This can be visually seen from figure 6

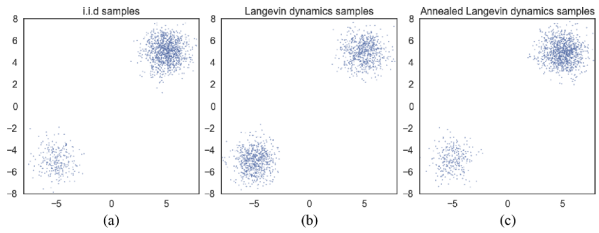

Figure 6: Samples from a mixture of Gaussian with different methods. (a) Exact sampling. (b) Sampling using Langevin dynamics with exact scores. (c) Sampling with annealed Langevin dynamics with the exact scores

2.4 Noise Conditional Score Networks

Possible solution to the problem.
* Use multiple σ noise scale!

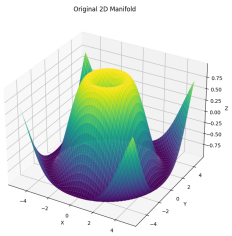

Figure 7: Original manifold

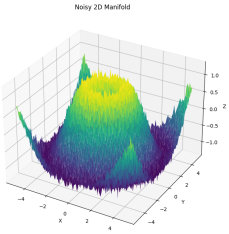

Figure 8: Gaussian perturbed manifold

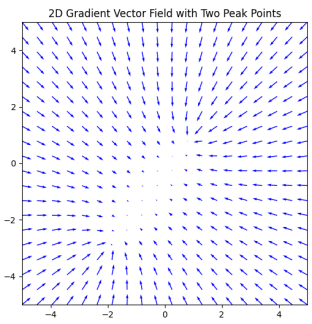

Figure 9: Original gradient field

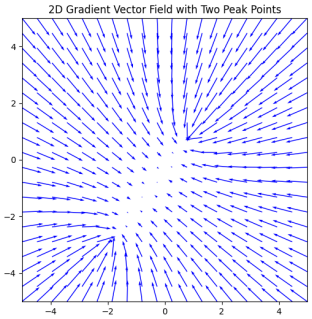

Figure 10: Step size scaled gradient field

Rather than using one gradient map and scaling by ϵ, we can train multiple gradient map with different σ_t scales. This helps us cover more *ambient space* as seen on figure 7 and 8. Also, we can get out of this space more easily in any step of training by using different gradient map as in figure **??** rather than just different sclae as in figure 10.

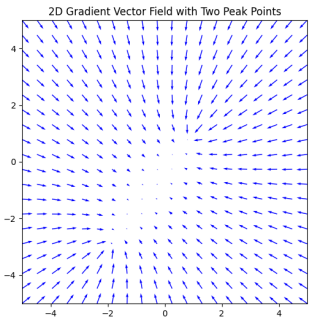

Figure 11: Original gradient field

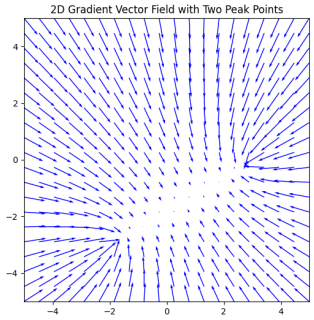

Figure 12: Gaussian perturbed gradient field

Then a loss (score matching function) for single σ is same as before.
Let noise distribution be $q_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) = \mathcal{N}(\tilde{\mathbf{x}}|\mathbf{x}, \sigma^2 I)$

$$\nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) = -\frac{\tilde{\mathbf{x}} - \mathbf{x}}{\sigma^2}$$

$$l(\theta; \sigma) := \frac{1}{2}\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\mathbb{E}_{\tilde{\mathbf{x}}\sim\mathcal{N}(\mathbf{x},\sigma^2 I)}\left[||s_\theta(\tilde{\mathbf{x}}, \sigma) + \frac{\tilde{\mathbf{x}} - \mathbf{x}}{\sigma^2}||_2^2\right](\forall\sigma \in \{\sigma_i\}_{i=1}^L)$$

$$\mathcal{L}(\theta; \{\sigma_i\}_{i=1}^L) := \frac{1}{L}\sum_{i=1}^{L} \lambda(\sigma_i) l(\theta; \sigma_i)(\lambda(\sigma_i) > 0, \quad \forall i \in [1, \ldots, L])$$

Empirically, it is known from Song et. al [9] that, $||s_\theta(\mathbf{x}, \sigma)||_2 \propto \frac{1}{\sigma}$ So to let the scales of loss across different σ same, we simply chose $\lambda(\sigma) = \sigma^2$

$$\lambda(\sigma) l(\theta; \sigma) = \sigma^2 l(\lambda; \sigma)$$

$$= \frac{1}{2}\mathbb{E}[||\sigma s_\theta(\tilde{\mathbf{x}}, \sigma) + \frac{\tilde{\mathbf{x}} - \mathbf{x}}{\sigma}||_2^2]$$

$$||\sigma s_\theta(\mathbf{x}, \sigma)||_2 \propto 1 \text{ and } \frac{\tilde{\mathbf{x}} - \mathbf{x}}{\sigma} \sim \mathcal{N}(0, I)$$

$$\therefore \lambda(\sigma) l(\theta; \sigma) \text{ does not depend on } \sigma$$

2.4.1 NCSN inference via annealed Langevien dynamics [9]

Algorithm 1 Annealed Langevin dynamics

Require: $\{\sigma_i\}_{i=1}^{L}, \epsilon, T$.
Initialize $\tilde{\mathbf{x}}_0$
for $i \leftarrow 1$ to L **do**
  $\alpha_i \leftarrow \epsilon \cdot \sigma_i^2/\sigma_L^2$
  for $t \leftarrow 1$ to T **do**
    Draw $\mathbf{z}_t \sim \mathcal{N}(0, I)$
    $\tilde{\mathbf{x}}_t \leftarrow \tilde{\mathbf{x}}_{t-1} + \frac{\alpha_i}{2}\mathbf{s}_\theta(\tilde{\mathbf{x}}_{t-1}, \sigma_i) + \sqrt{\alpha_i}\mathbf{z}_t$
  end for
  $\tilde{\mathbf{x}}_0 \leftarrow \tilde{\mathbf{x}}_T$
end for
return $\tilde{\mathbf{x}}_T$

Motivation of step size α_i

$$\text{When } \alpha_i \propto \sigma_i^2,$$

From the update "signal" from the algorithm above,

$$\tilde{\mathbf{x}}_t \leftarrow \tilde{\mathbf{x}}_{t-1} + \underbrace{\frac{\alpha_i}{2}\mathbf{s}_\theta(\tilde{\mathbf{x}}_{t-1}, \sigma_i)}_{\text{Signal Term}} + \underbrace{\sqrt{\alpha_i}\mathbf{z}_t}_{\text{Noise Term}}$$

Signal to Noise Ratio (SNR) is,

$$\mathbb{E}[||\frac{\alpha_i s_\theta(\mathbf{x}, \sigma_i)}{2\sqrt{\alpha_i}\mathbf{z}}||_2^2] \approx \mathbb{E}[\frac{\alpha_i ||s_\theta(\mathbf{x}, \sigma_i)||^2}{4}] \propto \frac{1}{4}\mathbb{E}[||\sigma_i s_\theta(\mathbf{x}, \sigma_i)||_2^2]$$

Since $\mathbb{E}[||\sigma_i s_\theta(\mathbf{x}; \sigma_i)||_2^2] \propto 1$,

$$\frac{1}{4}\mathbb{E}[||\sigma_i s_\theta(\mathbf{x}, \sigma_i)||_2^2] \propto \frac{1}{4}$$

. So the SNR is fixed for every σ_i.

2.4.2 Choice of σ_i

$\{\sigma_i\}_{i=1}^{L}$ is chosen as a positive geometric sequence that satisfies,

$$\frac{\sigma_1}{\sigma_2} = \cdots = \frac{\sigma_{L-1}}{\sigma_L} > 1$$

In the experienment of the original paper [9], σ_is were chosen so that, $\sigma_1 = 1$ and $\sigma_{10} = 0.01$

Limitation [10]

- Low resolution (32×32)
- Long computation time

2.4.3 Improved NCSN [10]

Technique 1: σ_1 depends on the data. Choose the initial noise scale σ_1 to be as large as the maximum Euclidean distance between all pairs of training data points.

Technique 2: (Other noise scales). Choose $\{\sigma_i\}_{i=1}^{L}$ as a geometric progression with common ration γ, such that $\Phi(\sqrt{2D}(\gamma-1)+3\gamma)-\Phi(\sqrt{2D}(\gamma-1)-3\gamma) \approx 0.5$ (D is dimension of image)

Technique 3: Parameterize the NCSN with $s_\theta(\mathbf{x}, \sigma) = s_\theta(\mathbf{x})/\sigma$, where $s_\theta(\mathbf{x})$ is an unconditional score network.

Technique 4: (Selecting T and ϵ). Choose T as large as allowed by computing budget and then select an ϵ that makes Eq. (4) of [10] maximally close to 1.

Technique 5: (EMA). Apply exponential moving average to parameters when sampling.

3 Diffusion

Idea

- We use multi-σ_i noise to learn different scales of noises, which makes it less flexible to wide range of data in high dimensions.
- Can the σ_is be gradual changes to distribution rather than just applying to original distribution?
- Original idea from non-equilibrium statistical physics. [4] and sequential Monte Carlo [6]
- First diffusion model idea paper is Jascha et al. [8]

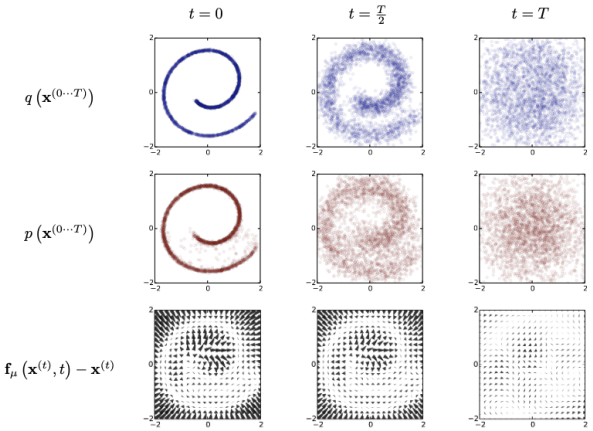

Figure 13: The proposed modeling from the original diffusion paper [8] trained on 2-d swiss roll data.

The *diffusion* or *forward* process maps a data example $\mathbf{x}$ through a Gaussian noise (left to right). And the *reverse* process which maps series of latent variables to the original data.

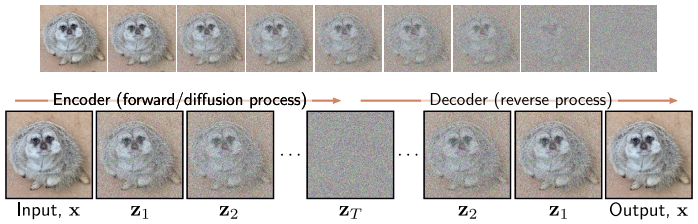

Figure 14: Diffusion overview

3.1 Encoder (forward process)

Note. The proofs here are from Simon J. D. Prince Understanding Deep Learning. [7]

In the *forward process*, data example $\mathbf{x}$ is mapped to intermediate variables $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n$ with $\beta_t \in [0, 1]$ known as *noise schedule* which determines how quickly the noise is mixed.

$$\mathbf{z}_1 = \sqrt{1-\beta_1} \cdot \mathbf{x} + \sqrt{\beta_1} \cdot \epsilon_1$$

$$\mathbf{z}_t = \sqrt{1-\beta_t} \cdot \mathbf{z}_{t-1} + \sqrt{\beta_t} \cdot \epsilon_t \qquad \forall t \in \{2, \ldots, T\}$$

$$\text{Where, } \epsilon_i \in \mathcal{N}(\mathbf{0}, \mathbf{I})$$

Let the encoder model q which does not have any learning parameters.

$$q(\mathbf{z}_1|\mathbf{x}) = \mathcal{N}_{\mathbf{z}_1}(\sqrt{1-\beta_1}\mathbf{x}, \beta_1\mathbf{I})$$

$$q(\mathbf{z}_t|\mathbf{z}_{t-1}) = \mathcal{N}_{\mathbf{z}_t}(\sqrt{1-\beta_t}\mathbf{z}_{t-1}, \beta_t\mathbf{I}) \qquad \forall t \in \{2, \ldots, T\}$$

This sequence of change of distribution is *Markov chain* because the probability $\mathbf{z}_t$ only depends on the value of the immediate preceding variable $\mathbf{z}_{t-1}$. And with the sufficient steps T, all traces of the original signal data are removed and left with just standard normal distribution.

$$q(\mathbf{z}_T|\mathbf{x}) = q(\mathbf{z}_T)$$

$$q(\mathbf{z}_{1\ldots T}|\mathbf{x}) = q(\mathbf{z_1}|\mathbf{x}) \prod_{t=2}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1})$$

This forward process can be seen on figure 15

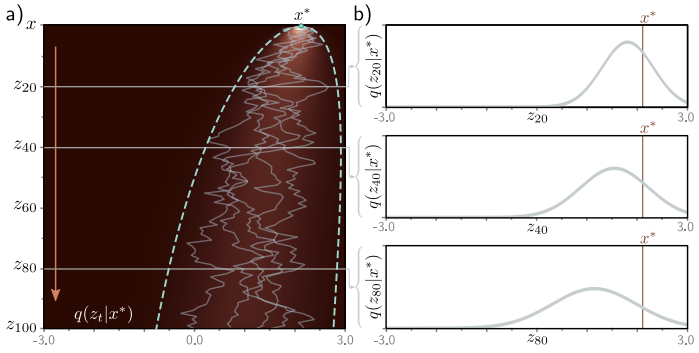

Figure 15: Forward diffusion process

Lets look into the forward diffusion step more.

$$\mathbf{z}_1 = \sqrt{1-\beta_1} \cdot \mathbf{x} + \sqrt{\beta_1} \cdot \epsilon_1$$
$$\mathbf{z}_2 = \sqrt{1-\beta_2} \cdot \mathbf{z}_1 + \sqrt{\beta_2} \cdot \epsilon_2$$
$$\mathbf{z}_2 = \sqrt{1-\beta_2}\sqrt{1-\beta_1} \cdot \mathbf{x} + \sqrt{\beta_1}\sqrt{1-\beta_2} \cdot \epsilon_1 + \sqrt{\beta_2} \cdot \epsilon_2$$
$$= \sqrt{(1-\beta_1)(1-\beta_2)} \cdot \mathbf{x} + \sqrt{1-(1-\beta_1)(1-\beta_2)} \cdot \epsilon$$

Lemma 1.

$\sqrt{\beta_1}\sqrt{1-\beta_2}\cdot\epsilon_1 + \sqrt{\beta_2}\cdot\epsilon_2 \approx \sqrt{1-(1-\beta_1)(1-\beta_2)}\cdot\epsilon$ *(from same distribution)*

Recall covariance matrix definition.

$$\Sigma = \mathbb{E}[(\mathbf{x}-\mu)(\mathbf{x}-\mu)^T]$$

$$\mathbb{E}[(\sqrt{\beta_1}\sqrt{1-\beta_2} \cdot \epsilon_1 + \sqrt{\beta_2} \cdot \epsilon_2 - \mathbf{0})(\sqrt{\beta_1}\sqrt{1-\beta_2} \cdot \epsilon_1 + \sqrt{\beta_2} \cdot \epsilon_2 - \mathbf{0})^T]$$
$$= \mathbb{E}[\beta_1(1-\beta_2)\epsilon_1\epsilon_1^T + \beta_2\epsilon_2\epsilon_2^T + \sqrt{\beta_1(1-\beta_2)(\beta_2)}\epsilon_1\epsilon_2^T + \sqrt{\beta_1(1-\beta_2)(\beta_2)}\epsilon_2\epsilon_1^T]$$
$$= \beta_1(1-\beta_2)\mathbb{E}[\epsilon_1\epsilon_1^T] + \beta_2\mathbb{E}[\epsilon_2\epsilon_2^T] + \sqrt{\beta_1(1-\beta_2)(\beta_2)}\mathbb{E}[\epsilon_1\epsilon_2^T] + \sqrt{\beta_1(1-\beta_2)(\beta_2)}\mathbb{E}[\epsilon_2\epsilon_1^T]$$

And since ϵ_1, ϵ_2 *are independent,* $\mathbb{E}[\epsilon_1\epsilon_2^T] = \mathbb{E}[\epsilon_1]\mathbb{E}[\epsilon_2^T] = \mathbf{0}$

$$= (\beta_1(1-\beta_2) + \beta_2)\mathbf{I} = (1-(1-\beta_1)(1-\beta_2))\mathbf{I}$$

$$\epsilon \sim \mathcal{N}(0, (1-(1-\beta_1)(1-\beta_2))\mathbf{I})$$

$\therefore \sqrt{\beta_1}\sqrt{1-\beta_2}\cdot\epsilon_1 + \sqrt{\beta_2}\cdot\epsilon_2 \approx \sqrt{1-(1-\beta_1)(1-\beta_2)}\cdot\epsilon$ *(from same distribution)*

□

By applying the same logic,

$$\mathbf{z}_t = \sqrt{(1-\beta_t)\dots(1-\beta_1)} \cdot \mathbf{x} + \sqrt{1-(1-\beta_t)\dots(1-\beta_1)} \cdot \epsilon$$

Let,

$$\alpha_t = \prod_{s=1}^{t} 1-\beta_s,$$
$$\mathbf{z}_t = \sqrt{\alpha_t} \cdot \mathbf{x} + \sqrt{1-\alpha_t} \cdot \epsilon$$
$$q(\mathbf{z}_t|\mathbf{x}) = \mathcal{N}_{\mathbf{z}_t}(\sqrt{\alpha_t} \cdot \mathbf{x}, (1-\alpha_t)\mathbf{I})$$

3.1.1 Marginal distributions $q(\mathbf{z}_t)$

$$q(\mathbf{z}_t) = \int q(\mathbf{z}_t|\mathbf{x})Pr(\mathbf{x})d\mathbf{x}$$

However, since we don't know the original data distribution this cannot be calculated.
Then how about reversing the process?

3.1.2 Conditional distributions $q(\mathbf{z}_{t-1}|\mathbf{z}_t)$

$$q(\mathbf{z}_{t-1}|\mathbf{z}_t) = \frac{q(\mathbf{z}_t|\mathbf{z}_{t-1})q(\mathbf{z}_{t-1})}{q(\mathbf{z}_t)}$$

This is intractalbe as well since we cannot calculate $q(\mathbf{z}_{t-1})$

3.1.3 Conditional diffusion distribution $q(\mathbf{z}_{t-1}|\mathbf{z}_t, \mathbf{x})$

However by using the starting condition $\mathbf{x}$ we can get tractable conditional distribution in closed form.

$$\begin{aligned} q(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}) &= \frac{q(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x})q(\mathbf{z}_{t-1}|\mathbf{x})}{q(\mathbf{z_t}|\mathbf{x})} \\ &\propto q(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x})q(\mathbf{z}_{t-1}|\mathbf{x}) \\ &= \mathcal{N}_{\mathbf{z}_t}(\sqrt{1-\beta_t}\cdot\mathbf{z}_{t-1}, \beta_t\mathbf{I})\mathcal{N}_{\mathbf{z}_{t-1}}(\sqrt{\alpha_{t-1}}\cdot\mathbf{x}, (1-\alpha_{t-1})\mathbf{I}) \end{aligned}$$

Lemma 2. *Gaussian change of variables identity,*

$$\mathcal{N}_{\mathbf{v}}(A\mathbf{w}, \mathbf{B}) \propto \mathcal{N}_{\mathbf{w}}((A^TB^{-1}A)^{-1}A^TB^{-1}\mathbf{v}, (A^TB^{-1}A)^{-1})$$

For general case, I will prove if asked.
invertible A case

$$\mathcal{N}_{\mathbf{v}}(A\mathbf{w}, \mathbf{B}) = \frac{1}{(2\pi)^d B^{1/2}}\exp((\mathbf{v}-A\mathbf{w})^TB^{-1}(\mathbf{v}-A\mathbf{w}))$$

$$= \frac{1}{(2\pi)^d B^{1/2}}\exp((A(A^{-1}\mathbf{v}-\mathbf{w}))^TB^{-1}(A(A^{-1}\mathbf{v}-\mathbf{w})))$$

$$\frac{1}{(2\pi)^d B^{1/2}}\exp((A^{-1}\mathbf{v}-\mathbf{w})^TA^{-T}B^{-1}A(A^{-1}\mathbf{v}-\mathbf{w}))$$

$$\mathcal{N}_{\mathbf{v}}(A\mathbf{w}, \mathbf{B}) \propto \mathcal{N}_{\mathbf{w}}, (A^{-1}\mathbf{v}, (A^TB^{-1}A)^{-1})$$

□

$$= \mathcal{N}_{\mathbf{z}_{t-1}}(\frac{1}{\sqrt{1-\beta_t}}\cdot\mathbf{z}_t, \frac{\beta_t}{1-\beta_t}\mathbf{I})\mathcal{N}_{\mathbf{z}_{t-1}}(\sqrt{\alpha_{t-1}}\cdot\mathbf{x}, (1-\alpha_{t-1})\mathbf{I})$$

Lemma 3.

$$\mathcal{N}_{\mathbf{w}}(\mathbf{a}, A) \cdot \mathcal{N}_{\mathbf{w}}(\mathbf{b}, B) \propto \mathcal{N}_{\mathbf{w}}((A^{-1} + B^{-1})^{-1}(A^{-1}\mathbf{a} + B^{-1}\mathbf{b}), (A^{-1} + B^{-1})^{-1})$$

$$\mathcal{N}_{\mathbf{w}}(\mathbf{a}, A) \cdot \mathcal{N}_{\mathbf{w}}(\mathbf{b}, B) \propto \exp((\mathbf{w} - \mathbf{a})^T A^{-1}(\mathbf{w} - \mathbf{a}) + (\mathbf{w} - \mathbf{b})^T B^{-1}(\mathbf{w} - \mathbf{b}))$$

Lets approximate this by using only 1st, 2nd degree term w.r.t $\mathbf{w}$

$$\approx \exp(\mathbf{w}^T A^{-1}\mathbf{w} + \mathbf{w}^T B^{-1}\mathbf{w} - (\mathbf{a}^T A^{-1} + \mathbf{b}^T B^{-1})\mathbf{w} - \mathbf{w}^T(A^{-1}\mathbf{a} + \mathbf{B}^{-1}\mathbf{b}))$$

$$= \exp((\mathbf{w} - \square)^T (A^{-1} + B^{-1})(\mathbf{w} - \square)$$

$$= \exp((\mathbf{w} - (A^{-1}+B^{-1})^{-1}(A^{-1}\mathbf{a}+B^{-1}\mathbf{b}))^T (A^{-1}+B^{-1})(\mathbf{w} - (A^{-1}+B^{-1})^{-1}(A^{-1}\mathbf{a}+B^{-1}\mathbf{b}))$$

$$\mathcal{N}_{\mathbf{w}}(\mathbf{a}, A) \cdot \mathcal{N}_{\mathbf{w}}(\mathbf{b}, B) \propto \mathcal{N}_{\mathbf{w}}((A^{-1} + B^{-1})^{-1}(A^{-1}\mathbf{a} + B^{-1}\mathbf{b}), (A^{-1} + B^{-1})^{-1})$$

□

$$= \mathcal{N}_{\mathbf{z}_{t-1}}(\frac{1}{\sqrt{1-\beta_t}} \cdot \mathbf{z}_t, \frac{\beta_t}{1-\beta_t}\mathbf{I})\mathcal{N}_{\mathbf{z}_{t-1}}(\sqrt{\alpha_{t-1}} \cdot \mathbf{x}, (1-\alpha_{t-1})\mathbf{I})$$

$$= \mathcal{N}_{\mathbf{z}_{t-1}}(\frac{(1-\alpha_{t-1})}{1-\alpha_t}\sqrt{1-\beta_t}\mathbf{z}_t + \frac{\sqrt{\alpha_{t-1}}\beta_t}{1-\alpha_t}\mathbf{x}, \frac{\beta_t(1-\alpha_{t-1})}{1-\alpha_t}\mathbf{I})$$

3.1.4 Decoder model (reverse process)

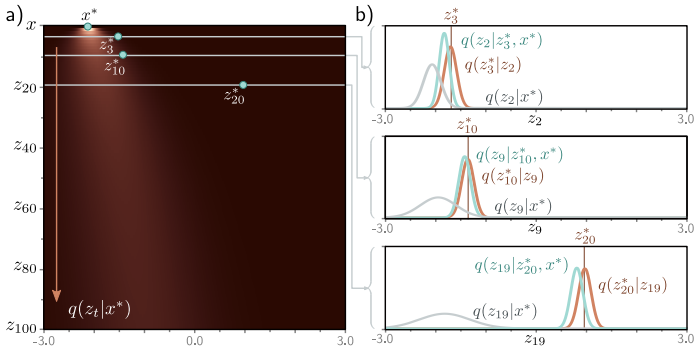

Figure 16: Backward diffusion process

As seen on figure 16 the true $q(\mathbf{z}_{t-1}|\mathbf{z}_t)$ of the diffusion process are complex multi-modal distributions. We approximate this as normal distributions.

$$\Pr(\mathbf{z}_T) = \mathcal{N}_{\mathbf{z}_T}(\mathbf{0}, \mathbf{I})$$
$$\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t, \phi_t) = \mathcal{N}_{\mathbf{z}_{t-1}}(f_t(\mathbf{z}_t, \phi_t), \sigma_t^2\mathbf{I})$$
$$\Pr(\mathbf{x}|\mathbf{z}_1, \phi_1) = \mathcal{N}_{\mathbf{x}}(f_1(\mathbf{z}_1, \phi_1), \sigma_1^2\mathbf{I})$$

3.1.5 Training

To train directly, we need to maximize,

$$\Pr(\mathbf{x}, \mathbf{z}_{1\ldots N}|\phi_{1\ldots N}) = \Pr(\mathbf{x}|\mathbf{z}_1, \phi_1)\prod_{t=2}^{T}\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t, \phi_t)\cdot\Pr(\mathbf{z}_T)$$

$$\Pr(\mathbf{x}|\phi_{1\ldots T}) = \int \Pr(\mathbf{x}, \mathbf{z}_{1\ldots T}|\phi_{1\ldots T})d\mathbf{z}_{1\ldots T}$$

$$\hat{\phi}_{1\ldots T} = \arg\max_{\phi_{1\ldots T}}[\sum_{i=1}^{I}\log[\Pr(\mathbf{x_i}|\phi_{\mathbf{1}\ldots\mathbf{T}})]]$$

But this is intractable because we cannot integrate over $\mathbf{z}_{1\ldots T}$

3.1.6 Evidence lower bound(ELBO)

Hence, we use Jensen's inequality to get evidence lower bound.

$$\log[\Pr(\mathbf{x}|\phi_{\mathbf{1}\ldots\mathbf{T}})] = \log[\int \Pr(\mathbf{x}, \mathbf{z}_{\mathbf{1}\ldots\mathbf{T}}|\phi_{\mathbf{1}\ldots\mathbf{T}})d\mathbf{z}_{1\ldots T}]$$

$$= \log[\int q(\mathbf{z}_{1\ldots T}|\mathbf{x})\frac{\Pr(\mathbf{x}, \mathbf{z}_{1\ldots T}|\phi_{1\ldots T})}{q(\mathbf{z}_{1\ldots T}|\mathbf{x})}d\mathbf{z}_{1\ldots T}]$$

$$\geq \int q(\mathbf{z}_{1\ldots T}|\mathbf{x})\log[\frac{\Pr(\mathbf{x}, \mathbf{z}_{1\ldots T}|\phi_{1\ldots T})}{q(\mathbf{z}_{1\ldots T}|\mathbf{x})}]d\mathbf{z}_{1\ldots T}$$

$$= \text{ELBO}[\phi_{1\ldots T}]$$

$$\log[\frac{\Pr(\mathbf{x}, \mathbf{z}_{1\ldots T}|\phi_{1\ldots T})}{q(\mathbf{z}_{1\ldots T}|\mathbf{x})}] = \log[\frac{\Pr(\mathbf{x}|\mathbf{z}_1, \phi_1)\prod_{t=2}^{T}\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t, \phi_t)\cdot\Pr(\mathbf{z}_T)}{q(\mathbf{z}_1|\mathbf{x})\prod_{t=2}^{T}q(\mathbf{z_t}|\mathbf{z_{t-1}})}]$$

$$= \log[\frac{\Pr(\mathbf{x}|\mathbf{z}_1, \phi_1)}{q(\mathbf{z}_1|\mathbf{x})}] + \log[\frac{\prod_{t=2}^{T}\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t, \phi_t)}{\prod_{t=2}^{T}q(\mathbf{z}_t|\mathbf{z}_{t-1})}] + \log[\Pr(\mathbf{z}_T)]$$

And we know from before,

$$q(\mathbf{z}_t|\mathbf{z}_{t-1}) = q(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}) = \frac{q(\mathbf{z}_{t-1}|\mathbf{z}_t, \mathbf{x})q(\mathbf{z}_t|\mathbf{x})}{q(\mathbf{z}_{t-1}|\mathbf{x})}$$

$$\log[\frac{\Pr(\mathbf{x}|\mathbf{z}_1,\phi_1)}{q(\mathbf{z}_1|\mathbf{x})}]+\log[\frac{\prod_{t=2}^{T}\Pr(\mathbf{z}_{t-1}|\mathbf{z}_r,\phi_t)}{\prod_{t=2}^{T}q(\mathbf{z}_t|\mathbf{z}_{t-1})}]+\log[\Pr(\mathbf{z}_T)]$$

$$=\log[\frac{\Pr(\mathbf{x}|\mathbf{z}_1,\phi_1)}{q(\mathbf{z}_1|\mathbf{x})}]+\log[\frac{\prod_{t=2}^{T}\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t,\phi_t)}{\prod_{t=2}^{T}q(\mathbf{z}_{t-1}|\mathbf{z}_t,\,\mathbf{x})}\cdot\frac{q(\mathbf{z}_{t-1}|\mathbf{x})}{q(\mathbf{z}_t|\mathbf{x})}]+\log[\Pr(\mathbf{z}_T)]$$

$$=\log[\Pr(\mathbf{x}|\mathbf{z}_1,\phi_1)]+\log[\frac{\prod_{t=2}^{T}\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t,\phi_t)}{\prod_{t=2}^{T}q(\mathbf{z}_{t-1}|\mathbf{z}_t,\,\mathbf{x})}]+\log[\frac{\Pr(\mathbf{z}_T)}{q(\mathbf{z}_T|\mathbf{x})}]$$

$$\approx\log[\Pr(\mathbf{x}|\mathbf{z}_1,\phi_1)]+\sum_{t=2}^{T}\log[\frac{\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t,\phi_t)}{q(\mathbf{z}_{t-1}|\mathbf{z}_t,\,\mathbf{x})}]$$

$$\mathrm{ELBO}[\phi_{1\ldots T}]=\int q(\mathbf{z}_{1\ldots T}|\mathbf{x})\log[\frac{\Pr(\mathbf{x},\mathbf{z}_{1\ldots T}|\phi_{1\ldots T})}{q(\mathbf{z}_{1\ldots T}|\mathbf{x})}]d\mathbf{z}_{1\ldots T}$$

$$\approx\int q(\mathbf{z}_{1\ldots T}|\mathbf{x})\bigg(\log[\Pr(\mathbf{x}|\mathbf{z}_1,\phi_1)]+\sum_{t=2}^{T}\log[\frac{\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t,\phi_t)}{q(\mathbf{z}_{t-1}|\mathbf{z}_t,\,\mathbf{x})}]\bigg)d\mathbf{z}_{1\ldots T}$$

$$=\mathbb{E}_{q(\mathbf{x}|\mathbf{z}_{1*},\phi_{1*})}[\log[\Pr(\mathbf{x}|\mathbf{z}_1,\phi_1)]]-\sum_{t=2}^{T}\mathbb{E}_{q(\mathbf{z}_t|\mathbf{x})}\Big[D_{KL}[q(\mathbf{z}_{t-1}|\mathbf{z}_t,\mathbf{x})|\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t,\phi_t)]\Big]$$

Lemma 4. *KL divergence between two normal distribution.*

$$D_{KL}[\mathcal{N}(\mathbf{a},A)|\mathcal{N}(\mathbf{b},B)]=\frac{1}{2}(tr[B^{-1}A]-d+(\mathbf{a}-\mathbf{b})^TB^{-1}(\mathbf{a}-\mathbf{b})+\log[\frac{|B|}{|A|}])$$

Proof:

$$D_{KL}[\mathcal{N}(\mathbf{a},A)|\mathcal{N}(\mathbf{b},B)]=\int\mathcal{N}(\mathbf{a},A)\log\bigg(\frac{(2\pi)^{d/2}|B^{-1/2}|}{(2\pi)^{d/2}|A^{-1/2}|}\exp(-(\mathbf{x}-\mathbf{a})^TA^{-1}(\mathbf{x}-\mathbf{a})+(\mathbf{x}-\mathbf{b})^TB^{-1}(\mathbf{x}-\mathbf{b})\bigg)d\mathbf{x}$$

$$=\int\mathcal{N}(\mathbf{a},A)-\frac{1}{2}\log\bigg(\frac{|B|}{|A|}\bigg)d\mathbf{x}+\int\mathcal{N}(\mathbf{a},A)\,(-(\mathbf{x}-\mathbf{a})^TA^{-1}(\mathbf{x}-\mathbf{a})+(\mathbf{x}-\mathbf{b})^TB^{-1}(\mathbf{x}-\mathbf{b})\bigg)d\mathbf{x}$$

$$=-\frac{1}{2}\log\bigg(\frac{|B|}{|A|}\bigg)+\int\mathcal{N}(\mathbf{a},A)\bigg((\mathbf{x}^T(B^{-1}-A^{-1})\mathbf{x}+2\mathbf{a}^TA^{-1}\mathbf{x}-2\mathbf{b}^TB^{-1}\mathbf{x}-\mathbf{a}^TA^{-1}\mathbf{a}+\mathbf{b}^TB^{-1}\mathbf{b})\bigg)d\mathbf{x}$$

$$=\frac{1}{2}\log\bigg(\frac{|A|}{|B|}\bigg)+\mathbb{E}_{\mathcal{N}(\mathbf{a},A)}[(\mathbf{x}^T(B^{-1}-A^{-1})\mathbf{x}+2\mathbf{a}^TA^{-1}\mathbf{x}-2\mathbf{b}^TB^{-1}\mathbf{x}-\mathbf{a}^TA^{-1}\mathbf{a}+\mathbf{b}^TB^{-1}\mathbf{b})]$$

$$=\frac{1}{2}\log\bigg(\frac{|A|}{|B|}\bigg)+\mathbb{E}_{\mathcal{N}(\mathbf{a},A)}[tr\bigg(\mathbf{x}^T(B^{-1}-A^{-1})\mathbf{x}\bigg)]+2\mathbf{a}^TA^{-1}\mathbf{a}-2\mathbf{b}^TB^{-1}\mathbf{a}-\mathbf{a}^TA^{-1}\mathbf{a}+\mathbf{b}^TB^{-1}\mathbf{b})$$

$$=\frac{1}{2}\log\bigg(\frac{|A|}{|B|}\bigg)+\mathbb{E}_{\mathcal{N}(\mathbf{a},A)}[tr\bigg(\mathbf{x}^T(B^{-1}-A^{-1})\mathbf{x}\bigg)]+\mathbf{a}^TA^{-1}\mathbf{a}-2\mathbf{b}^TB^{-1}\mathbf{a}+\mathbf{b}^TB^{-1}\mathbf{b})$$

$$=\frac{1}{2}\log\bigg(\frac{|A|}{|B|}\bigg)+\mathbb{E}_{\mathcal{N}(\mathbf{a},A)}[tr\bigg((B^{-1}-A^{-1})\mathbf{x}\mathbf{x}^T\bigg)]+\mathbf{a}^TA^{-1}\mathbf{a}-2\mathbf{b}^TB^{-1}\mathbf{a}+\mathbf{b}^TB^{-1}\mathbf{b})$$

$$= \frac{1}{2}\log\left(\frac{|A|}{|B|}\right) + tr\left((B^{-1} - A^{-1})(A + \mathbf{a}\mathbf{a}^{\mathbf{T}})\right) + \mathbf{a}^T A^{-1}\mathbf{a} - 2\mathbf{b}^T B^{-1}\mathbf{a} + \mathbf{b}^T B^{-1}\mathbf{b})$$

$$= \frac{1}{2}\log\left(\frac{|A|}{|B|}\right) + tr\left(B^{-1}A + \mathbf{I}\right) + \mathbf{a}^T B^{-1}\mathbf{a} - 2\mathbf{b}^T B^{-1}\mathbf{a} + \mathbf{b}^T B^{-1}\mathbf{b})$$

$$== \frac{1}{2}(tr[B^{-1}A] - d + (\mathbf{a} - \mathbf{b})^T B^{-1}(\mathbf{a} - \mathbf{b}) + \log[\frac{|B|}{|A|}])$$

□

$$= \mathbb{E}_{\text{ıı}(\mathbf{x}|\mathbf{z}_{\text{ȷ}\kappa}, \phi_{\text{ȷ}\kappa})}[\log[\Pr(\mathbf{x}|\mathbf{z}_1, \phi_1)]] - \sum_{t=2}^{T} \mathbb{E}_{q(\mathbf{z}_t|\mathbf{x})}\Big[D_{KL}[q(\mathbf{z}_{t-1}|\mathbf{z}_t, \mathbf{x})|\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t, \phi_t)]\Big]$$

$$Pr(\mathbf{z}_{t-1}|\mathbf{z}_t, \phi_t) = \mathcal{N}_{\mathbf{z_{t-1}}}[f_t[\mathbf{z}_t, \phi_t], \sigma_t^2\mathbf{I}]$$

$$q(\mathbf{z_{t-1}}|\mathbf{z_t}, \mathbf{x}) = \mathcal{N}_{\mathbf{z}_{t-1}}[\frac{(1-\alpha_{t-1})}{1-\alpha_t}\sqrt{1-\beta_t}\mathbf{z}_t + \frac{\sqrt{\alpha_{t-1}}\beta_t}{1-\alpha_t}\mathbf{x}, \frac{\beta_t(1-\alpha_{t-1})}{1-\alpha_t}\mathbf{I}]$$

$$D_{KL}[q(\mathbf{z}_{t-1}|\mathbf{z}_t, \mathbf{x})|\Pr(\mathbf{z}_{t-1}|\mathbf{z}_t, \phi_t)]$$

$$= \frac{1}{2\sigma_t^2}||\frac{(1-\alpha_{t-1})}{1-\alpha_t}\sqrt{1-\beta_t}\mathbf{z}_t + \frac{\sqrt{\alpha_{t-1}}\beta_t}{1-\alpha_t}\mathbf{x} - f_t[\mathbf{z}_t, \phi_t]||^2 + C$$

$$L[\phi_{1\ldots T}] = \sum_{i=1}^{I}\Bigg(\underbrace{-\log[\mathcal{N}_{\mathbf{x}_i}[f_1[\mathbf{z}_{i1}, \phi_1], \sigma_1^2\mathbf{I}]]}_{\text{reconstruction term}} + \sum_{t=2}^{T}\frac{1}{2\sigma_t^2}||\underbrace{\frac{1-\alpha_{t-1}}{1-\alpha_t}\sqrt{1-\beta_t}\mathbf{z}_{it} + \frac{\sqrt{\alpha_{t-1}}\beta_t}{1-\alpha_t}\mathbf{x}_i}_{\text{target, mean of } q(\mathbf{z_{t-1}}|\mathbf{z_t}, \mathbf{x})} - \underbrace{f_t[\mathbf{z}_{it}, \phi_t]}_{\text{predicted } \mathbf{z}_{t-1}}||\Bigg)$$

3.1.7 Reparmeterization trick

$$\mathbf{z}_t = \sqrt{\alpha_t}\cdot\mathbf{x} + \sqrt{1-\alpha_t}\cdot\epsilon$$

$$\mathbf{x} = \frac{1}{\sqrt{\alpha_t}}\cdot\mathbf{z}_t - \frac{\sqrt{1-\alpha_t}}{\sqrt{\alpha_t}}\cdot\epsilon$$

$$\frac{1-\alpha_{t-1}}{1-\alpha_t}\sqrt{1-\beta_t}\mathbf{z}_t + \frac{\sqrt{\alpha_{t-1}}\beta_t}{1-\alpha_t}\mathbf{x}$$

$$= \frac{(1-\alpha_{t-1})}{1-\alpha_t}\sqrt{1-\beta_t}\mathbf{z}_t + \frac{\sqrt{\alpha_{t-1}}\beta_t}{1-\alpha_t}\left(\frac{1}{\sqrt{\alpha_t}}\mathbf{z}_t - \frac{\sqrt{1-\alpha_t}}{\sqrt{\alpha_t}}\epsilon\right)$$

$$= \frac{(1-\alpha_{t-1})}{1-\alpha_t}\sqrt{1-\beta_t}\mathbf{z}_t + \frac{\beta_t}{1-\alpha_t}\left(\frac{1}{\sqrt{1-\beta_t}}\mathbf{z}_t - \frac{\sqrt{1-\alpha_t}}{\sqrt{1-\beta_t}}\epsilon\right)$$

$$= \left(\frac{(1-\alpha_{t-1})\sqrt{1-\beta_t}}{1-\alpha_t} + \frac{\beta_t}{(1-\alpha_t)\sqrt{1-\beta_t}}\right)\mathbf{z}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}\sqrt{1-\beta_t}}\epsilon$$

$$= \left(\frac{(1-\alpha_{t-1})(1-\beta_t)}{(1-\alpha_t)\sqrt{1-\beta_t}} + \frac{\beta_t}{(1-\alpha_t)\sqrt{1-\beta_t}}\right)\mathbf{z}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}\sqrt{1-\beta_t}}\epsilon$$

$$= \left(\frac{(1-\alpha_{t-1})(1-\beta_t)+\beta_t}{(1-\alpha_t)\sqrt{1-\beta_t}} \right) \mathbf{z}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}\sqrt{1-\beta_t}} \epsilon$$

$$= \left(\frac{1-\alpha_t}{(1-\alpha_t)\sqrt{1-\beta_t}} \right) \mathbf{z}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}\sqrt{1-\beta_t}} \epsilon$$

$$= \left(\frac{1}{\sqrt{1-\beta_t}} \right) \mathbf{z}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}\sqrt{1-\beta_t}} \epsilon$$

$$\therefore L[\phi_{1\ldots T}] = \sum_i^I \Bigg(-\log[\mathcal{N}_{\mathbf{x}_i}[f_1[\mathbf{z}_{i1}, \phi_1], \sigma_1^2 \mathbf{I}]]$$

$$+ \sum_{t=2}^T \frac{1}{2\sigma_t^2} || (\frac{1}{\sqrt{1-\beta_t}} \mathbf{z}_{it} - \frac{\beta_t}{\sqrt{1-\alpha_t}\sqrt{1-\beta_t}} \epsilon_{it}) - f_t[\mathbf{z}_{it}, \phi_t]||^2 \Bigg)$$

When we replace the model $\hat{\mathbf{z}}_{t-1} = f_t[\mathbf{z}_t, \phi_t]$ with $\hat{\epsilon} = g_t[\mathbf{z}_t, \phi_t]$ which is predicting noise mixed to the $\mathbf{x}$,

$$f_t[\mathbf{z}_t, \phi_t] = \frac{1}{\sqrt{1-\beta_t}} \mathbf{z}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}\sqrt{1-\beta_t}} g_t[\mathbf{z}_t, \phi_t]$$

$$L[\phi_{1\ldots T}] = \sum_{i=1}^I \left(-\log[\mathcal{N}_{\mathbf{x_i}}[f_1[\mathbf{z}_{i1}, \sigma_1^2 \mathbf{I}]]] + \sum_{t=2}^T \frac{\beta_t^2}{(1-\alpha_t)(1-\beta_t) 2\sigma_t^2} ||g_t[\mathbf{z}_{it}, \phi_t] - \epsilon_{it}||^2 \right)$$

$$= \sum_{i=1}^I \left(\frac{1}{2\sigma_1^2} ||\mathbf{x}_i - f_1[\mathbf{z}_{i1}, \phi_1]||^2 + \sum_{t=2}^T \frac{\beta_t^2}{(1-\alpha_t)(1-\beta_t) 2\sigma_t^2} ||g_t[\mathbf{z}_{it}, \phi_t] - \epsilon_{it}||^2 + C_i \right)$$

$$\frac{1}{2\sigma_1^2} ||\mathbf{x}_i - f_1[\mathbf{z}_{i1}, \phi_1]||^2 = \frac{1}{2\sigma^2} || \frac{\beta_1}{\sqrt{1-\alpha_1}\sqrt{1-\beta_1}} g_1[\mathbf{z}_{i1}, \phi_1] - \frac{\beta_1}{\sqrt{1-\alpha_1}\sqrt{1-\beta_1}} \epsilon_{i1} ||$$

The loss can be reduced to

$$L[\phi_{1\ldots T}] = \sum_{i=1}^I \sum_{t=1}^T \frac{\beta_t^2}{(1-\alpha_t)(1-\beta_t) 2\sigma_t^2} ||g_t[\mathbf{z}_{it}, \phi_t] - \sigma_{it}]||^2$$

Scaling factors can also be ignored

$$L[\phi_{1\ldots T}] = \sum_{i=1}^I \sum_{t=1}^T ||g_t[\mathbf{z}_{it}, \phi_t] - \sigma_{it}]||^2$$

$$= \sum_{i=1}^I \sum_{t=1}^T ||g_t[\sqrt{\alpha_t} \cdot \mathbf{x}_i + \sqrt{1-\alpha_t} \cdot \epsilon_{it}, \phi_t] - \sigma_{it}]||^2$$

3.2 Algorithms

Algorithm 2 Diffusion model training

Input: Training data $\mathbf{x}$
Output: Model parameters ϕ_t
repeat
  for $i \in \mathcal{B}$ **do**
    $t \sim \text{Uniform}[1, \ldots, T]$
    $\epsilon \sim \mathcal{N}[\mathbf{0}, \mathbf{I}]$
    $l_i = ||g_t[\sqrt{\alpha_t}\mathbf{x}_i + \sqrt{1-\alpha_t}\epsilon, \phi_t] - \epsilon||^2$
  end for
  Accumulate losses for batch and take gradient step
until converged

Algorithm 3 Diffusion model sampling

Input: Model, $g_t[\cdot, \phi_t]$
Output: generated sample, $\mathbf{x}$
$\mathbf{z}_T \sim \mathcal{N}_{\mathbf{z}}[\mathbf{0}, \mathbf{I}]$
for $t = T \ldots 2$ **do**
  $\hat{\mathbf{z}}_{t-1} = \frac{1}{\sqrt{1-\beta_t}}\mathbf{z}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}\sqrt{1-\beta_t}} g_t[\mathbf{z}_t, \phi_t]$
  $\epsilon \sim \mathcal{N}_{\epsilon}[\mathbf{0}, \mathbf{I}]$
  $\mathbf{z}_{t-1} = \hat{\mathbf{z}}_{t-1} + \sigma_t \epsilon$
end for
$\mathbf{x} = \frac{1}{\sqrt{1-\beta_1}}\mathbf{z}_1 - \frac{\beta_1}{\sqrt{1-\alpha_1}\sqrt{1-\beta_1}} g_1[\mathbf{z_1}, \phi_\mathbf{1}]$

Due to time constraint, denoising diffusion implicit models will be covered next time if needed

Appendix

A)

$$\mathcal{N}(\mu, \Sigma) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp(-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu))$$

Proof: [2]

let $\mathbf{x} \sim \mathcal{N}_0(\mathbf{0}, I)$ be i.i.d multi Gaussian distribution with $\mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}$ and

$$x_i \sim \mathcal{N}(0,1) = \frac{1}{(2\pi)^{1/2}} \exp(-\frac{1}{2}x^2) \ (\forall i \in [1,n])$$

$$p(x_1, x_2, \ldots, x_n) = p(x_1|x_2, x_3, \ldots, x_n)p(x_2|x_3, x_4, \ldots, x_n)\ldots p(x_n)$$

because of i.i.d assumption,

$$= p(x_1)p(x_2)\ldots p(x_n)$$

$$= \frac{1}{(2\pi)^{n/2}} \exp(-\frac{1}{2}(x_1^2 + x_2^2 + \cdots + x_n^2))$$

$$= \frac{1}{(2\pi)^{n/2}} \exp(\mathbf{x}^T\mathbf{x})$$

Let new correlated distribution $\mathbf{y} = \Sigma^{1/2}\mathbf{x} + \mu$

$$\frac{\delta \mathbf{y}}{\delta \mathbf{x}} d\mathbf{x} = d\mathbf{y}$$

$$d\mathbf{y} = \Sigma^{1/2} d\mathbf{x}$$

$$\frac{\delta \mathbf{y}}{\delta \mathbf{x}} = \Sigma^{1/2}$$

If we only consider the volume change,

$$d\mathbf{y} = |\det(\Sigma^{1/2})| d\mathbf{x}$$

$$p(\mathbf{x})d\mathbf{x} = p(\mathbf{x}(\mathbf{y}))\frac{\delta \mathbf{x}}{\delta \mathbf{y}} d\mathbf{y}$$

$$\mathbf{x} = \Sigma^{-1/2}(\mathbf{y} - \mu)$$

$$= \frac{1}{(2\pi)^{n/2}} \exp((\Sigma^{-1/2}(\mathbf{y}-\mu))^T(\Sigma^{-1/2}(\mathbf{y}-\mu)))\frac{1}{|\det(\Sigma^{1/2})|}$$

$$= \frac{1}{(2\pi)^{n/2}|\Sigma^{1/2}|} \exp\left((\mathbf{y}-\mu)^T\Sigma^{-1}(\mathbf{y}-\mu)\right)$$

□

References

[1] Alexander N. Gorban and Ivan Y. Tyukin. Blessing of dimensionality: mathematical foundations of the statistical physics of data. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2118):20170237, April 2018.

[2] Robert V. Hogg, Joseph W. McKean, and Allen T. Craig. *Mathematical Statistics with Applications*. Cengage Learning, Boston, MA, 9th edition, 2020.

[3] Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005.

[4] C. Jarzynski. Equilibrium free-energy differences from nonequilibrium measurements: A master-equation approach. *Physical Review E*, 56(5):5018–5035, November 1997.

[5] Eamonn Keogh and Abdullah Mueen. Curse of dimensionality. In Claude Sammut and Geoffrey I. Webb, editors, *Encyclopedia of Machine Learning*, pages 257–258. Springer, Boston, MA, 2011.

[6] Radford M. Neal. Annealed importance sampling, 1998.

[7] Simon J.D. Prince. *Understanding Deep Learning*. MIT Press, 2023.

[8] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015.

[9] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution, 2020.

[10] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *CoRR*, abs/2006.09011, 2020.