Understanding DDPM

by Cristian Lazo Quispe

Author: Cristian Lazo Quispe

Bayesian Inference

(design choice) generative model → likelihood

(design choice) what we assume about z before seeing data → prior

$$\text{posterior} \rightarrow p(z|x) = \frac{p(x|z)\,p(z)}{p(x)}$$

Observed Data: x (images, exam scores)

Latent Variable: z (label, intelligence)

given image infer label

evidence (marginal likelihood)

$p(x) = \int p(x,z)\,dz = \int p(x|z)\,p(z)\,dz$ Intractable part

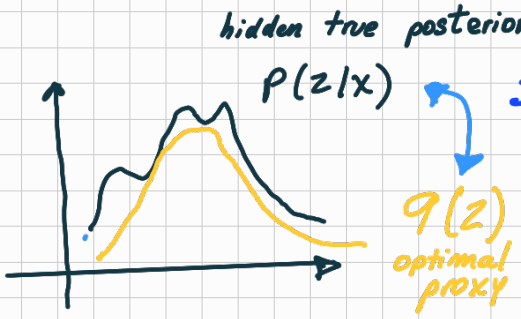
(The integral is over high-dimensional space)



We use approximations like ELBO (variational inference) to compute $p(z|x)$

Variational Inference

Method of approximating a complex distribution with some family of parameterized distributions.



Divergence

$KL[q(z) \| p(z|x)]$

quantify the closeness

We need to find a distribution which is "close" to the true posterior

$$KL[q(z) \| p(z|x)] = \int q(z) \log \frac{q(z)}{p(z|x)}\,dz = \mathop{E}_{z\sim q(z)}\left[\log \frac{q(z)}{p(z|x)}\right]$$

However, the spaces of search is infinitive, what do we do instead?

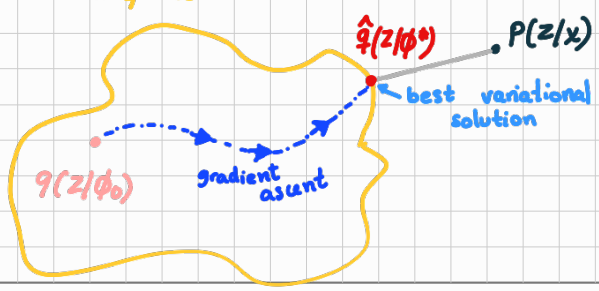
We could restrict our search to a manageable family of distributions $\mathbb{Q}$, like:

$q(\theta|\phi) = \mathcal{N}(\mu_\phi, \sigma^2_\phi)$ we just optimize the parameters: μ_ϕ σ^2_ϕ

Thus, variational inference attempts to find the optimal $\hat{q}$

$$\hat{q} = \arg\min_{q} KL[q(z) \| p(z|x)]$$

Space of all $q: \mathbb{Q}$



↓ this is a hard problem to solve



We can solve it using ELBO

Author: Cristian Lazo Quispe

$$\hat{q} = \arg\min_{q} \; KL\left[q(z) \,\|\, p(z|x)\right]$$

$$\phi^* = \arg\min_{\phi} \; KL\left[q(z|\phi) \,\|\, p(z|x)\right]$$

$$\phi^* = \arg\min_{\phi} \; E_{z\sim q(z|\phi)}\left[\log\left(\frac{q(z|\phi)}{p(z|x)}\right)\right]$$

$$\phi^* = \arg\min_{\phi} \; E_{z\sim q(z|\phi)}\left[\log\left(\frac{q(z|\phi)}{p(z,x)/p(x)}\right)\right]$$

$$\phi^* = \arg\min_{\phi} \; E_{z\sim q(z|\phi)}\left[\log(q(z|\phi)) - \left[\log(p(z,x)) - \log p(x)\right]\right]$$

$$\phi^* = \arg\min_{\phi} \; E_{z\sim q(z|\phi)}\left[\log q(z|\phi) - \log(p(z,x)) + \log p(x)\right]$$

$$\phi^* = \arg\min_{\phi} \; E_{z\sim q(z|\phi)}(\log(p(x))) + E_{z\sim q(z|\phi)}\left[\log q(z|\phi) - \log p(z,x)\right] \rightarrow E_{z\sim q(z|\phi)}\left[\log\left(\frac{q(z|\phi)}{q(z,x)}\right)\right]$$

$$\phi^* = \arg\min_{\phi} \; \log(p(x)) + E_{z\sim q(z|\phi)}\left[\log q(z|\phi)\right] - E_{z\sim q(z|\phi)}\left[\log p(z,x)\right]$$

$$\phi^* = \arg\min_{\phi} \; \underbrace{\log p(x)}_{\text{this not depend on } \phi} - \underbrace{\left(E_{z\sim q(z|\phi)}\left[\log p(z,x)\right] - E_{z\sim q(z|\phi)}\left[\log q(z|\phi)\right]\right)}_{\text{this is called ELBO}}$$

evidence lower bound

$$\phi^* = \arg\min_{\phi} \; -ELBO(\phi)$$

$$\phi^* = \arg\max_{\phi} \; ELBO(\phi)$$



Now the optimization is in terms of ELBO

We know that

$$KL(q(z) \,\|\, p(z|x)] \geq 0$$

$$\log p(x) - ELBO(\phi) \geq 0$$

$$\log p(x) \geq ELBO(\phi)$$



ELBO is a lower bound

$$ELBO = \mathbb{E}_{z\sim q(z|\phi)} \log p(z,x) - \mathbb{E}_{z\sim q(z|\phi)} \log q(z|\phi)$$

$$ELBO = \mathbb{E}_{z\sim q(z|\phi)} \log p(x|z)\cdot p(z) - \mathbb{E}_{z\sim q(z|\phi)} \log q(z|\phi)$$

$$ELBO = \mathbb{E}_{z\sim q(z|\phi)} \log p(x|z) + \mathbb{E}_{z\sim q(z|\phi)} \log p(z) - \mathbb{E}_{z\sim q(z|\phi)} \log q(z|\phi)$$

$$ELBO = \mathbb{E}_{z\sim q(z|\phi)} \log p(x|z) - \mathbb{E}_{z\sim q(z|\phi)} [\log q(z|\phi) - \log p(z)]$$

remember

$KL(A,B) \neq KL(B,A)$

$$ELBO = \mathbb{E}_{z\sim q(z|\phi)} \log p(x|z) - \mathbb{E}_{z\sim q(z|\phi)} \left[\log \frac{q(z|\phi)}{p(z)}\right]$$

usually prior define as $N(0,I)$

$N(\mu_\phi, \sigma^2_\phi)$

$$ELBO = \underbrace{\mathbb{E}_{z\sim q(z|\phi)} \log p(x|z)}_{\text{log likelihood}} - \underbrace{KL(q(z|\phi) \,||\, p(z))}_{\text{penalize weird latent space}}$$

it could be MSE loss



Now the optimization is computable

How to approximate the posterior $p(z|x)$:

$$\hat{q}(z) \approx p(z|x)$$

$$\hat{q} = \arg\min_q KL[q(z) \,||\, p(z|x)]$$

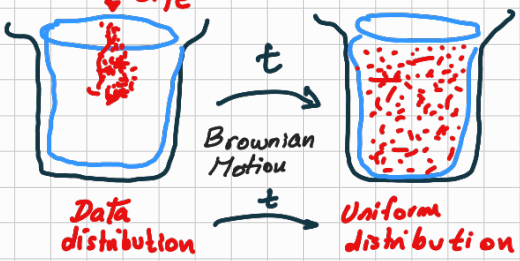
$$\phi^* = \arg\max_\phi ELBO(\phi)$$

$$ELBO = \text{log-likelihood} - KL(q(z|\phi) \,||\, p(z))$$

Deep Understanding learning Using Non equilibrium Thermodynamics

[ICML 2015 presentation]

Observation 1 Diffusion destroys structure



Goal: Learn structure of probability density

Dye density represents probability density

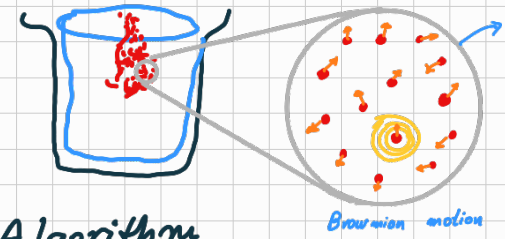
Observation: Diffusion destroys structure



Imagine what you could do if you could reverse time?

Data distribution ← running dynamics backward — Uniform distribution

Observation 2 Microscopic Diffusion is Time Reversible.



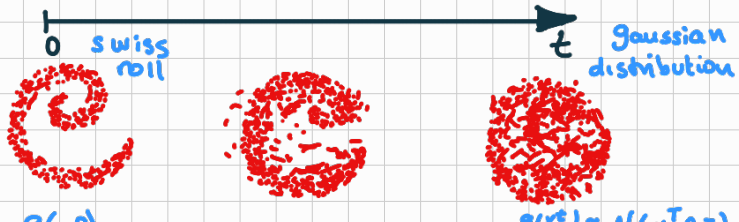
Position updates are small gaussians (next location around current location)

↳ reverse trajectory also consists of drawing a gaussian centered in the previous data point

Algorithm

1. Destroy all structure in data distribution using diffusion process
2. Train deep supervised network to reverse diffusion process
3. Reverse diffusion process constitutes generative model of data

Run Gaussian Diffusion



$q(x^0)$ $\quad q(x^t) \approx N(x^t; 0, I)$

$$q(x^t/x^{t-1}) = N(x^t; x^{t-1}\sqrt{1-\beta_t}, I\beta_t)$$

Decay towards origin — add small noise

β_t is learnable

Reversal of Diffusion Process

$$p(x^0) \approx q(x^0) \leftarrow p(x^t) = N(x^t; 0, I)$$

$$p(x^{t-1}/x^t) = N(x^{t-1}; f_\mu(x^t, t), f_\Sigma(x^t, t))$$

mean and covariance functions defined using deep supervised learning

Training

$$\underset{f_\mu(x^t,t),\, f_\Sigma(x^t,t)}{\operatorname{argmin}} \; \mathbb{E}\left[D_{KL}\left(q(x^{t-1}/x^t, x^0) \,\|\, p(x^{t-1}/x^t)\right)\right]$$

R gaussians t



the model learns the conditional transition $P_\theta(x^{t-1}/x^t)$ directly

Fit functions to known means and covariances of sequence of gaussians

Author: Cristian Lazo Quispe

Denoising Diffusion Probabilistic Models

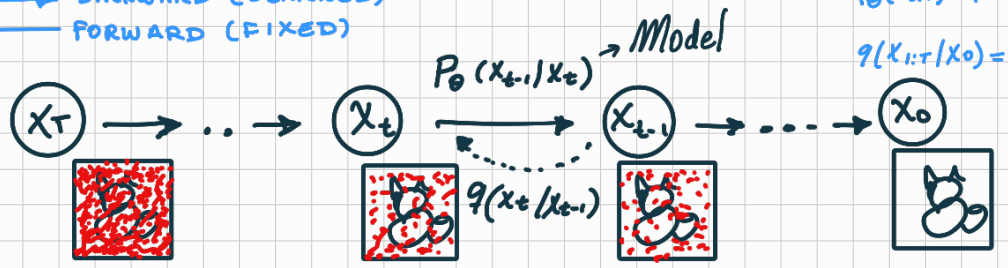


What if, instead of learning the distribution of how data transitions $p_\theta(x^{t-1}/x^t)$, we simply learn the noise that was added at each step?

→ BACKWARD (LEARNED)
← FORWARD (FIXED)

$$p_\theta(x_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}/x_t)$$

$$q(x_{1:T}/x_0) = \prod_{t=1}^{T} q(x_t/x_{t-1})$$



$p_\theta(x_{0:T})$ is called the reverse process, and it is defined as a Markov chain with learned Gaussian transitions starting at $p(x_T) = N(x_T; 0, I)$

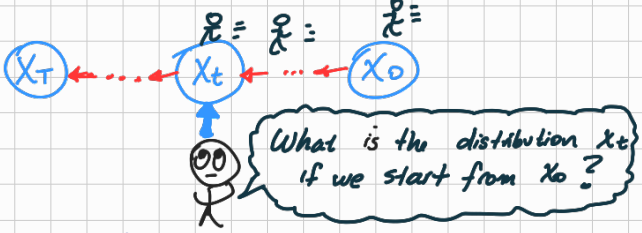
$$p_\theta(x_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}/x_t)$$

,

$$p_\theta(x_{t-1}/x_t) = N(x_{t-1}; \mu_\theta(x_t,t), \Sigma_\theta(x_t,t))$$

(model)

1. FORWARD PROCESS $q(x_t/x_0)$



lets use this notation: $\alpha_t = 1-\beta_t$ $\quad \bar{\alpha}_t = \prod_{s=1}^{t}\alpha_s$ $\quad \varepsilon_t \sim N(0, I)$

β_t is fixed parameter

$$q(x_1/x_0) = N(x_1, x_0\sqrt{\alpha_1}, I(1-\alpha_1))$$

$$x_1 = \sqrt{\alpha_1}\,x_0 + \sqrt{1-\alpha_1}\,\varepsilon_1$$

$$x_2 = \sqrt{\alpha_2}\,x_1 + \sqrt{1-\alpha_2}\,\varepsilon_2$$

$$\begin{cases} \sqrt{\alpha_2}\left[\sqrt{\alpha_1}\,x_0 + \sqrt{1-\alpha_1}\,\varepsilon_1\right) + \sqrt{1-\alpha_2}\,\varepsilon_2 \\ \sqrt{\alpha_2}\sqrt{\alpha_1}\,x_0 + \sqrt{\alpha_2}\sqrt{1-\alpha_1}\,\varepsilon_1 + \sqrt{1-\alpha_2}\,\varepsilon_2 \\ x_2 = \sqrt{\alpha_2}\sqrt{\alpha_1}\,x_0 + \sqrt{\alpha_2}\sqrt{1-\alpha_1}\,\varepsilon_1 + \sqrt{1-\alpha_2}\,\varepsilon_2 \end{cases}$$

$$x_3 = \sqrt{\alpha_3}\,x_2 + \sqrt{1-\alpha_3}\,\varepsilon_3$$

$$\begin{cases} \sqrt{\alpha_3}\left(\sqrt{\alpha_2}\sqrt{\alpha_1}\,x_0 + \sqrt{\alpha_2}\sqrt{1-\alpha_1}\,\varepsilon_1 + \sqrt{1-\alpha_2}\,\varepsilon_2\right) + \sqrt{1-\alpha_3}\,\varepsilon_3 \\ \sqrt{\alpha_3}\sqrt{\alpha_2}\sqrt{\alpha_1}\,x_0 + \sqrt{\alpha_3}\sqrt{\alpha_2}\sqrt{1-\alpha_1}\,\varepsilon_1 + \sqrt{\alpha_3}\sqrt{1-\alpha_2}\,\varepsilon_2 + \sqrt{1-\alpha_3}\,\varepsilon_3 \\ x_3 = \sqrt{\prod\alpha_i}\,x_0 + \sqrt{\alpha_3\alpha_2(1-\alpha_1) + \alpha_3(1-\alpha_2) + 1-\alpha_3}\;\varepsilon \\ \qquad \alpha_3\alpha_2 - \alpha_3\alpha_2\alpha_1 + \alpha_3 - \alpha_3\alpha_2 + 1 - \alpha_3 \\ x_3 = \sqrt{\prod\alpha_i}\,x_0 + \sqrt{1-\alpha_3\alpha_2\alpha_1}\;\varepsilon \\ x_3 = \sqrt{\prod\alpha_i}\,x_0 + \sqrt{1-\prod\alpha_i}\;\varepsilon \end{cases}$$

$aN + bN + cN \;\to\; \sqrt{a^2+b^2+c^2}\,N$

$$x_T = \sqrt{\prod\alpha_i}\,x_0 + \sqrt{1-\prod\alpha_i}\;\varepsilon$$

$$x_T = \sqrt{\bar{\alpha}_t}\,x_0 + \sqrt{1-\bar{\alpha}_t}\;\varepsilon$$

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

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

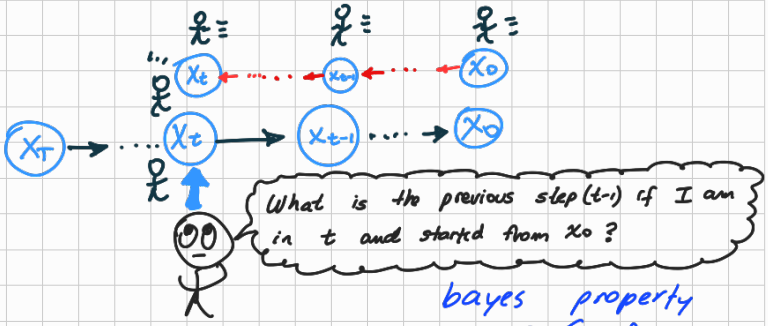
$$\boxed{x_t = \sqrt{\bar{\alpha}_t}\,x_0 + \sqrt{1-\bar{\alpha}_t}\;\varepsilon}$$

Author: Cristian Lazo Quispe

2. BACKWARD PROCESS

Model : $P_\theta(x_{t-1}/x_t)$

Ground Truth : $q(x_{t-1}/x_t, x_0)$



bayes property

$$= \frac{P(A \cap C) \cdot P(A \cap B \cap C)}{P(C) \; P(C \cap A)}$$

$$P(A, B/C) = P(A/C) \cdot P(B/C, A)$$

$$P(A, B/c) = P/B/C) \cdot P(A/C, B)$$

$$= \frac{q(x_{t-1}, x_t, x_0)}{q(x_t, x_0)} = \frac{q(x_{t-1}, x_t/x_0)\, q(x_0)}{q(x_t, x_0)}$$

$$= \frac{q(x_{t-1}, x_t/x_0)}{q(x_t/x_0)} = \frac{q(x_{t-1}/x_0)\, q(x_t/x_{t-1}, x_0)}{q(x_t/x_0)}$$

markov property

$x_t \perp x_0 / x_{t-1}$



$$q(x_t/x_{t-1}, x_0) = q(x_t/x_{t-1})$$

$$q(x_{t-1}/x_t, x_0) = \frac{q(x_{t-1}/x_0)\, q(x_t/x_{t-1})}{q(x_t/x_0)}$$

this term not depend on x_{t-1}

$$q(x_{t-1}/x_t, x_0) \propto q(x_{t-1}/x_0) \cdot q(x_t/x_{t-1})$$

$$q(x_{t-1}/x_t, x_0) \propto N(x_{t-1}, \sqrt{\bar{\alpha}_{t-1}}\, x_0, (1-\bar{\alpha}_{t-1})I) \cdot N(x_t, \sqrt{\alpha_t}\, x_{t-1}, (1-\alpha_t)I)$$

$$N(x_{t-1}, \sqrt{\bar{\alpha}_{t-1}}\, x_0, (1-\bar{\alpha}_{t-1})I) \cdot N(x_t, \sqrt{\alpha_t}\, x_{t-1}, (1-\alpha_t)I)$$

$$N(x_{t-1}, \mu_1, \sigma_1^2) \cdot N(\mu_2, \sqrt{\alpha_t}\, x_{t-1}, \sigma_2^2)$$

this is an identity

$$N(x, \mu_1, \sigma_1^2) \cdot N(\mu_2, ax, \sigma_2^2)$$

$$\propto N\left(x; \mu = \sigma^2\left(\frac{\mu_1}{\sigma_1^2} + a\frac{\mu_2}{\sigma_2^2}\right), \sigma^2 = \frac{\sigma_2^2 \sigma_1^2}{\sigma_2^2 + a^2\sigma_1^2}\right)$$

$$\sigma^2 = \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{(1-\alpha_t + \alpha_t(1-\bar{\alpha}_{t-1})}$$

$$\sigma^2 = \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{(1-\alpha_t + \alpha_t - \alpha_t\bar{\alpha}_{t-1})}$$

$\bar{\alpha}_t$

$$\sigma^2 = \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{(1-\bar{\alpha}_t)}$$

$$\mu = \sigma^2\left(\frac{\mu_1}{\sigma_1^2} + a\frac{\mu_2}{\sigma_2^2}\right)$$

$$\mu = \frac{(1-\bar{\alpha}_{t-1})}{(1-\bar{\alpha}_t)}(1-\alpha_t)\left[\frac{\sqrt{\bar{\alpha}_{t-1}}\, x_0}{(1-\bar{\alpha}_{t-1})} + \frac{\sqrt{\alpha_t}\, x_t}{(1-\alpha_t)}\right]$$

$$\mu = \frac{(1-\alpha_t)\sqrt{\bar{\alpha}_{t-1}}\, x}{(1-\bar{\alpha}_t)} + \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{(1-\bar{\alpha}_t)} x_t$$

$$q(x_{t-1}/x_t, x_0) = N(x_{t-1}; \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t I)$$

$$\tilde{\mu}_t(x_t, x_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\, \beta_t}{1-\bar{\alpha}_t} x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t$$

$$\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$

β_t is fixed parameter

$$q(x_{t-1}/x_t, x_0) = N(x_{t-1}, \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t I)$$

Author: Cristian Lazo Quispe

How do we learn $P_\theta(x_{t-1}/x_t)$ denoising $x_t \to x_{t-1}$

Loss: $-\log(P_\theta(x_0))$

$$-\log(P_\theta(x_0)) = -\log\int P(x_0/x_1)\,P(x_1/x_2)\,P(x_2/x_3)\dots P(x_{T-1}/x_T)\,P(x_T)\,dx_1 dx_2 \dots dx_T$$

$$-\log(P_\theta(x_0)) = -\log\int P_\theta(x_{0:T})\,dx_{1:T}$$

this is intractable → use Jense inequality

$$\log\int q(z)\,f(z)\,dz \geq \int q(z)\log f(z)\,dz$$

$$\log\int P_\theta(x_{0:T})\cdot\frac{q(x_{1:T}/x_0)}{q(x_{1:T}/x_0)}\,dx_{1:T} \geq \int q(x_{1:T}/x_0)\log\left(\frac{P_\theta(x_{0:T})}{q(x_{1:T}/x_0)}\right)dx_{1:T}$$

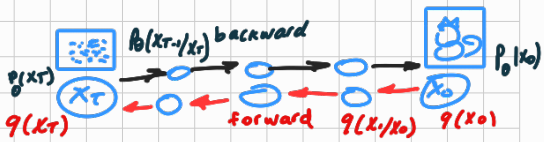
$$-\log\int P_\theta(x_{0:T})\,dx_{1:T} \leq -\int q(x_{1:T}/x_0)\log\left(\frac{P_\theta(x_{0:T})}{q(x_{1:T}/x_0)}\right)dx_{1:T}$$

$$-\log\int P_\theta(x_{0:T})\,dx_{1:T} \leq -\underset{q(x_{1:T}/x_0)}{E}\left[\log(P_\theta(x_{0:T})) - \log(q(x_{1:T}/x_0))\right]$$

$$-\log(P_\theta(x_0)) \leq -\underset{q(x_{1:T}/x_0)}{E}\left[\log(P_\theta(x_{0:T})) - \log(q(x_{1:T}/x_0))\right]$$

(computable)

- $\underset{q(x_{1:T}/x_0)}{E}\left[\log(P_\theta(x_{0:T})) - \log(q(x_{1:T}/x_0))\right]$



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

Join distribution over all steps

$P_\theta(x_0, x_1, \dots x_T)$ ($x_T \to \dots \to x_0$)

1. Starts from $x_T \sim N(0, I)$
2. Let's generate $x_{T-1}, x_{T-2}, \dots x_1, x_0$ using the learned model θ

$$q(x_{1:T}/x_0) = \prod_{t=1}^{T} q(x_t/x_{t-1}, x_0)$$

($x_T \leftarrow \dots \leftarrow x_0$)

Join distribution over 1:T steps given real data x_0

$$q(x_t/x_{t-1}) = N(x_t; \sqrt{\alpha_t}\,x_{t-1}, (1-\alpha_t)I)$$

1. starts from real data x_0
2. the noising process generates $x_1, x_2, \dots x_T$

ELBO

maximize $\underset{q(x_1:x_T/x_0)}{E}\left[\log\frac{p(x_0:x_T)}{q(x_1:x_T/x_0)}\right]$



$$\begin{cases} \log(P_\theta(x_{0:T})) = \log\left[P(x_T)\prod_{t=T}^{1} P_\theta(x_{t-1}/x_t)\right] = \log(P_\theta(x_T)) + \sum_{t=T}^{1}\log(P_\theta(x_{t-1}/x_t)) \\ \log(q(x_{1:T}/x_0)) = \log\left[\prod_{t=1}^{T} q(x_t/x_{t-1})\right] = \sum_{t=1}^{T}\log(q(x_t/x_{t-1})) \end{cases}$$

q depends on x_0

$$q(x_t/x_{t-1}) = \frac{q(x_{t-1}/x_t)\cdot q(x_t/x_0)}{q(x_{t-1}/x_0)}$$

- $E\left[\log(P_\theta(x_T)) + \sum_{t=T}^{1}\log(P_\theta(x_{t-1}/x_t) - \sum_{t=1}^{T}\log(q(x_t/x_{t-1}))\right]$

$$E\Big[\log(P_\theta(x_T)) + \log(P_\theta(x_0/x_1)) + \sum_{t=T}^{2}\log(P_\theta(x_{t-1}/x_t)) - \log q(x_1/x_0) - \sum\log\frac{q(x_{t-1}/x_t)\cdot q(x_t/x_0)}{q(x_{t-1}/x_0)}$$

$$-\log q(x_1/x_0) - \sum\log q(x_{t-1}/x_t) - \sum\log q(x_t/x_0) + \sum q(x_{t-1}/x_0)$$

(cancelled)

$$-\sum_{t=2}^{T}\log q(x_{t-1}/x_t) - \log q(x_T/x_0) + q(x_0/x_0)$$

(constant)

$$E\left[\log(P_\theta(x_T)) - \log(q(x_T/x_0))\right] + \sum_{t=2}^{T} E\left[\log(P_\theta(x_{t-1}/x_t)) - \log(q(x_{t-1}/x_t))\right] - E\left[\log(P_\theta(x_0/x_1))\right]$$

$$D_{KL}\left(q(x_T/x_0)\,\|\,P_\theta(x_T)\right) + \sum_{t=2}^{T} D_{KL}\left(q(x_{t-1}/x_t)\,\|\,P_\theta(x_{t-1}/x_t)\right) - E\left[\log(P_\theta(x_0/x_1))\right]$$

L_T — how gaussian is $P_\theta(x_T)$

L_{t-1} — how good is the generation in intermediate steps

L_0 — How good is the final reconstruction

1. $L_T = D_{KL}(q(x_T/x_0) \| p_\theta(x_T))$

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

$q(x_T/x_0)$ — Forward: $x_T \leftarrow \dots \leftarrow x_0$ — this is fixed (not learnable params)

$p_\theta(x_T)$ — Backward: $x_T \rightarrow$

we define x_T (starting point) $x_T \sim N(0, I)$

$p_\theta(x_T)$ this is fixed (not learnable params)

All the generation starts from noise x_T

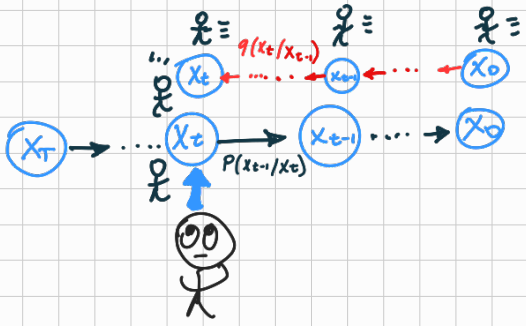
& $p_\theta(x_{t-1}/x_T)$ is learned

L_T is a constant during training and can be ignored

2. $L_{t-1} = D_{KL}(q(x_{t-1}/x_t) \| p_\theta(x_{t-1}/x_t))$

$$\tilde{\mu}_t(x_t, x_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t} x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t$$

$$\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$



$$q(x_{t-1}/x_t, x_0) = N(x_{t-1}, \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t I)$$

Model: $p_\theta(x_{t-1}/x_t)$ $\quad x_t \rightarrow x_{t-1}$

Ground Truth: $q(x_{t-1}/x_t, x_0)$ $\quad x_t \rightarrow x_{t-1}$

$$p_\theta(x_{t-1}/x_t) = N(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

- DDPM assumes that $\Sigma_\theta(x_t, t)$ is fixed $\sigma_t^2 I$ (2 options) $\begin{cases} \sigma_t^2 I = \beta_t \\ \sigma_t^2 I = \tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t \end{cases}$

∴ Only the mean function $\mu_\theta(x_t, t)$ should approximate the forward posterior mean of the true reverse process

$$\mu_\theta(x_t, t) \approx \tilde{\mu}_t(x_t, x_0)$$

$$L_{t-1} = \mathbb{E}_q\left[\frac{1}{2\sigma_t^2} \| \tilde{\mu}_t(x_t, x_0) - \mu_\theta(x_t, t)\|^2\right] + C$$

μ_θ is a model that predicts $\tilde{\mu}_t$

- $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\,\varepsilon$ we could express x_t as function of x_0, ε

$$L_{t-1} = \mathbb{E}_q\left(\frac{1}{2\sigma_t^2} \| \tilde{\mu}_t(x_t(x_0, \varepsilon), x_0) - \mu_\theta(x_t(x_0, \varepsilon), t)\|^2\right]$$

We could write L_{t-1} in terms of x_t

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\,\varepsilon \rightarrow \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1-\bar{\alpha}_t}\,\varepsilon) = x_0$$

Author: Cristian Lazo Quispe

$$L_{t-1} = \mathbb{E}_{x_0,\varepsilon}\left[\frac{1}{2\sigma_t^2}\left\| \underbrace{\tilde{\mu}_t\left(x_t(x_0,\varepsilon), \frac{1}{\sqrt{\bar{\alpha}_t}}\left(x_t - \sqrt{1-\bar{\alpha}_t}\,\varepsilon\right)\right)}_{} - \mu_\theta\left(x_t(x_0,\varepsilon), t\right)\right\|^2\right]$$

Now all terms depend on x_t, let's find an expression of $\tilde{\mu}_t(x_t, x_0)$ in terms of x_t

$$\boxed{\tilde{\mu}_t(x_t, x_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\,\beta_t}{1-\bar{\alpha}_t} x_0 + \frac{\sqrt{\alpha_t}\,(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t} \qquad \alpha_t = 1-\beta_t$$

$$\frac{\sqrt{\bar{\alpha}_{t-1}}\,\beta_t}{1-\bar{\alpha}_t}\left[\frac{1}{\sqrt{\bar{\alpha}_t}}\left(x_t - \sqrt{1-\bar{\alpha}_t}\,\varepsilon\right)\right] + \frac{\sqrt{\alpha_t}\,(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t -$$

$$\left[\frac{\sqrt{\bar{\alpha}_{t-1}}\,(1-\alpha_t)}{(1-\bar{\alpha}_t)\sqrt{\bar{\alpha}_t}\sqrt{\alpha_t}} + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{(1-\bar{\alpha}_t)}\right] x_t - \frac{\sqrt{\bar{\alpha}_{t-1}}\,\beta_t}{(1-\bar{\alpha}_t)}\frac{1}{\sqrt{\bar{\alpha}_t}}\sqrt{1-\bar{\alpha}_t}\,\varepsilon$$

$$\left[\frac{(1-\alpha_t) + \alpha_t(1-\bar{\alpha}_{t-1})}{(1-\bar{\alpha}_t)\sqrt{\alpha_t}}\right] x_t - \frac{\beta_t}{(1-\bar{\alpha}_t)}\frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\alpha_t}}\varepsilon$$

$$\left[\frac{1-\alpha_t + \alpha_t - \alpha_t\bar{\alpha}_{t-1}}{(1-\bar{\alpha}_t)\sqrt{\alpha_t}}\right] x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}\sqrt{\alpha_t}}\varepsilon$$

$$\left[\frac{1-\bar{\alpha}_t}{(1-\bar{\alpha}_t)\sqrt{\alpha_t}}\right] x_t - \frac{1}{\sqrt{\alpha_t}}\left(\frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\right)\varepsilon$$

$$\frac{1}{\sqrt{\alpha_t}} x_t - \frac{1}{\sqrt{\alpha_t}}\frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon$$

$$\frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon\right)$$

$$\tilde{\mu}_t(x_t, x_0) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon\right) \Big\} \tilde{\mu}_t(x_t)$$ let's replace in L_{t-1}

$$L_{t-1} = \mathbb{E}_{x_0,\varepsilon}\left\{\frac{1}{2\sigma_t^2}\left\|\frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon\right) - \mu_\theta\left(x_t(x_0,\varepsilon), t\right)\right\|^2\right\}$$



Since x_t is available as input to the model we may chose the parameterization

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t, t)\right)$$

where $\varepsilon_\theta(x_t, t)$ is a function approximator intended to predict ε from x_t, t

$$\therefore L_{t-1} = \mathbb{E}_{x_0,\varepsilon}\left[\frac{1}{2\sigma_t^2}\left[\frac{\beta_t^2}{\alpha_t(1-\bar{\alpha}_t)}\right]\left\|\varepsilon - \varepsilon_\theta\left(\sqrt{\bar{\alpha}_t}\,x_0 + \sqrt{1-\bar{\alpha}_t}\,\varepsilon,\ t\right)\right\|^2\right]$$



We can train it to predict ε

Author: Cristian Lazo Quispe

$$E\left[\log\left(\frac{p_\theta(x_T)}{q(x_T|x_0)}\right)\right] + \sum_{t=2}^{T} E\left[\log\left(\frac{p_\theta(x_{t-1}|x_t)}{q(x_{t-1}|x_0)}\right)\right] - E\left[\log(p_\theta(x_0|x_1))\right]$$

$D_{KL}(q(x_T|x_0) \| p_\theta(x_T))$ + $\sum_{t=2}^{T} D_{KL}(q(x_{t-1}|x_t) \| p_\theta(x_{t-1}|x_t))$ − $E[\log(p_\theta(x_0|x_1))]$

L_T — how gaussian is $p_\theta(x_T)$

L_{t-1} — how good is the generation in intermediate steps

L_0 — How good is the final reconstruction

$$L_{t-1} = \mathbb{E}_{x_0,\epsilon}\left[\frac{1}{2\sigma_t^2}\left[\frac{\beta_t^2}{\alpha_t(1-\bar{\alpha}_t)}\right]\left\| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\right\|^2\right]$$

D_{KL} dominates learning, L_T is constant, and L_0 is only needed for likelihood estimation (not for generation)



the noise prediction loss (MSE) is easy to optimize and give excellent results
It can be seen as a simplified, unweighted version of denoising score matching

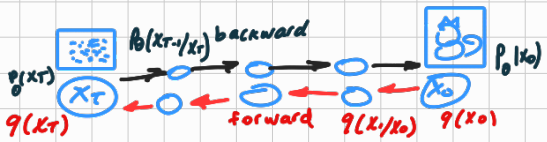
$t \sim \mathcal{U}(1,T)$

$$L_{simple}(\theta) := E_{t,x_0,\epsilon}\left[\left\| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + (\sqrt{1-\bar{\alpha}_t})\epsilon, t)\right\|^2\right]$$

SAMPLING:

BACKWARD

$$p_\theta(x_{t-1}|x_t) = N(x_{t-1}; \mu_\theta(x_t,t), \sigma_t^2 I)$$



$$x_{t-1} = \mu_\theta(x_t,t) + \sigma_t N(0,I)$$

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t,t)\right) + \sigma_t \underbrace{N(0,I)}_{z}$$

the input of the model expect to have noise and in last step ($t=1$) $z=0$

Now, we could implement the algorithms

ALGORITHM 1 Training

repeat
- $x_0 \sim q(x_0)$
- $t \sim$ Uniform $(\{1 \dots T\})$
- $\epsilon \sim N(0,I)$
- Take gradient descent step on $\nabla_\theta \| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\|^2$

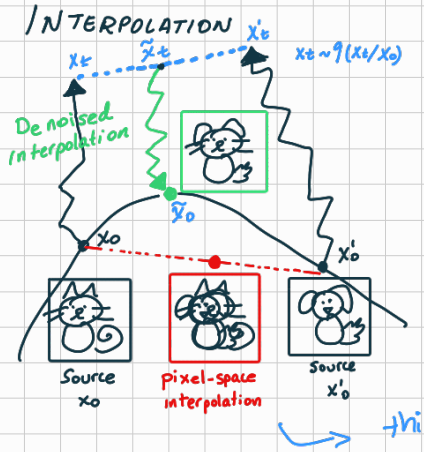
until converged

ALGORITHM 2 Sampling

$x_T \sim N(0,I)$
for $t = T, \dots, 1$ do
- $z \sim N(0,I)$ if $t>1$, else $z=0$
- $x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t,t)\right) + \sigma_t z$

end for
return x_0

INTERPOLATION



$\bar{x}_t = (1-\lambda)x_t + \lambda x'_t$ (interpolation in latent space)

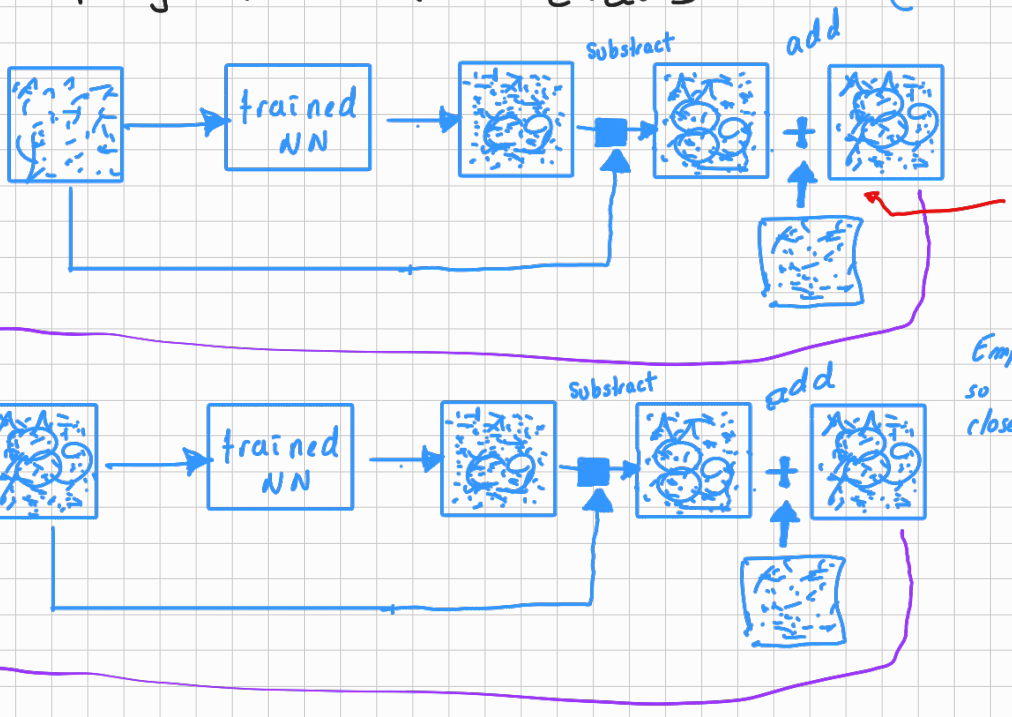
$\bar{x}_0 \sim p(x_0|\bar{x}_t)$

this not work because the pixel-space is not linearly smooth

Author: Cristian Lazo Quispe

Sampling Iteration Details

(Deeplearning.AI)



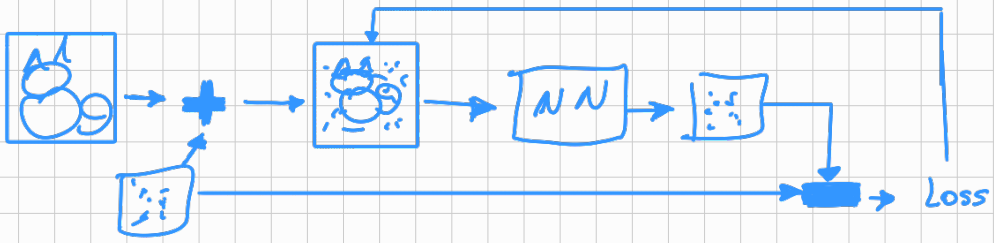
the NN expects a noisy sample as input

You can add in additional noise before it gets passed to the next step

Empirically, this stabilizes the NN so it doesn't collapse to something closer to the average of the dataset

Training

NN learns to predict noise - really learns the distribution of what is not noise. Sample random timestep (noise level) per image to train more stably



Author: Cristian Lazo Quispe