

# 完全理解数学推导非常难！！！！

## Diffusion probabilistic Model (扩散概率模型)

### 工作流程

1. 从高维正态分布采样，维度与生成图片的维度大小相同
2. 模型对采样后的数据去除噪声，会连续经过**去噪模型**（次数事先定好），最后把噪声通通滤掉，得到图片
  - Denoise模型除了会接受图片作为输入，还会接受一个数字作为参数，这个数字表示当前图片受噪声影响严重的程度，客观上反映了进行到哪个step
  - Denoise模型内部的**Noise Predictor**对图片噪声进行预测，得到噪声图片，然后用原图片减去噪声图片得到更加清晰的图片

 image-20250510233616851

### 数学推导

#### 马尔科夫分层自编码器

可以看作是VAE的一个扩展，从左到右逐步编码，从右到左逐步解码

 image-20250511132150467

$q(z_t|z_{t-1})$ 表示一个步骤的编码过程， $p(z_t|z_{t-1})$ 表示一个步骤的解码过程

整个模型的联合分布概率为  $p(x, z_1, z_2, \dots, z_T) = p(x, z_{1:T}) = p(z_T) p_{\theta}(x|z_1) \prod_{t=2}^T p_{\theta}(z_{t-1}|z_t)$

隐变量的 $z_{1:T}$ 后验概率可以分解为  $q_{\phi}(z_1, z_2, \dots, z_T | x) = q_{\phi}(z_{1:T}|x) = q_{\phi}(z_1|x) \prod_{t=2}^T q_{\phi}(z_t|z_{t-1})$  我们希望学到的模型能尽可能生成与真实样本分布一致的数据,模型的**优化目标**是为了**最大化** $p(x)$

$$\begin{aligned} \ln p(x) &= \ln \int p(x, z_{1:T}) dz_{1:T} \\ &= \ln \int \frac{p(x, z_{1:T}) q_{\phi}(z_{1:T}|x)}{q_{\phi}(z_{1:T}|x)} dz_{1:T} \\ &= \mathbb{E}_{q_{\phi}(z_{1:T}|x)} \left( \ln \frac{p(x, z_{1:T})}{q_{\phi}(z_{1:T}|x)} \right) \geq \end{aligned}$$

#### 扩散模型

前向编码过程的每一个步骤的编码器 $q(x_t|x_{t-1})$ 不再通过神经网络学习，而是固定为一个高斯线性变换。

不区分 $x$ 和 $z$ ，每个 $x_t$ 的尺寸都是相同的

由于编码器被假设为线性高斯，当 $T$ 趋向无穷时， $x_T$ 是一个正态分布即随着 $T$ 的增大， $x_T$ 趋近于正态分布。这个线性高斯设定一个小于 1 的渐系数，可以使 $x_T$ 收敛到一个标准正态分布

## 前向-后向

两种方法算出来的值是不同的

描述数据从干净状态 $x_0$ 逐步加噪到 $x_T$ 的路径概率  $p(x_{0:T}) = q(x_0)\prod_{t=1}^T q(x_t|x_{t-1})$

描述从噪声 $x_T$ 逐步去噪生成 $x_0$ 的路径概率  $p(x_{0:T}) = p(x_T)\prod_{t=T}^1 p(x_{t-1}|x_t)$

## 前向过程

代表真实图像 $x_0$ 的真实概率分布 $p(x_0)$ 的概率密度函数是不知道的，但我们能得到一批真实的图像样本，也就是我们有 $x_0$ 的观测样本，此时 $x_0$ 是已知观测值， $x_{1:T}$ 是未知的隐变量，这时整个马尔科夫网络的联合概率变成了一个条件概率 $q(x_{1:T}|x_0)$ 。 $q(x_{1:T}|x_0) = \frac{q(x_{0:T})}{q(x_0)} = \frac{q(x_0)\prod_{t=1}^T q(x_t|x_{t-1})}{q(x_0)} = \prod_{t=1}^T q(x_t|x_{t-1})$

根据前面的假设，前向过程每一个步骤的编码器 $q(x_t|x_{t-1})$ 固定为一个线性高斯变换。定义 $q(x_t|x_{t-1})$ 的方差与 $x_{t-1}$ 是独立的，并且为 $\beta_t$ ，其中 $0 < \beta_1 < \beta_2 < \dots < \beta_T < 1$ 。这么做的意义：前期方差较小，添加的噪声少，扩散速度慢；随着方差逐步加大，添加的噪声越来越多，扩散的速度加快。 $\beta_t$ 是人工指定的。

定义 $q(x_t|x_{t-1})$ 的均值 $\mu_{x_t}$ 和 $x_{t-1}$ 是线性关系，这里设定另外一个系数 $\alpha_t$ ，并且令 $\alpha_t = 1 - \beta_t$ 。 $\mu_t$ 与 $x_{t-1}$ 的关系定义为

- 这里是 $x_{t-1}$ 上的每一个像素点乘上一个 $\sqrt{\alpha_t}$ 得到每一个像素点的均值 $\mu_{x_t}$

$\mu_{x_t} = \sqrt{\alpha_t}x_{t-1}$   $x_t$ 的方差定义与 $x_{t-1}$ 无关，而是经过缩放的单位方差，定义为 $\Sigma_{x_t} = \beta_t I = (1 - \alpha_t)I$  那么这个时候 $q(x_t|x_{t-1})$ 就是一个以 $\sqrt{\alpha_t}x_{t-1}$ 为均值，以 $(1 - \alpha_t)I$ 为方差的高斯分布（方差固定不变）。它可以看作是在 $\sqrt{\alpha_t}x_{t-1}$ 的基础上加上一个 $\mathcal{N}(0, (1 - \alpha_t)I)$ 的随机高斯噪声。这就相当于每一个步骤都在前

一个步骤的基础上加上一个随机高斯噪声数据，随着  $t$  的增加， $x_t$  逐步变成一个高斯噪声数据。  $q(x_t|x_{t-1}) = \mathcal{N}(\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)I)$

$$\begin{aligned} x_t &= \sqrt{\alpha_t}x_{t-1} + \mathcal{N}(0, (1-\alpha_t)I) \\ &= \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\epsilon, \epsilon \sim \mathcal{N}(0, I) \end{aligned}$$

$\alpha_t$  并非固定，可以随着  $t$  的增长逐渐变小。

总的来说，前向过程就是一个逐步添加高斯噪声，最终变成一个纯高斯噪声数据的过程，无参数化表示，假定是一个确定的线性高斯变换。

前向过程中可以从  $x_0$  一步计算任意的  $t$ ，这样可以并行计算全部的  $x_t$ 。公式中的  $\epsilon_t$  是从一个标准正态分布中采样的。推导过程如下：

$$\begin{aligned} x_t &= \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\epsilon_t \\ &= (\sqrt{\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}}\epsilon_{t-1}) + \sqrt{1-\alpha_t}\epsilon_t \\ &= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \underbrace{\sqrt{\alpha_t(1-\alpha_{t-1})}\epsilon_{t-1} + \sqrt{1-\alpha_t}\epsilon_t}_{\text{两个相互独立的0均值的高斯分布相加}} \\ &= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \underbrace{\sqrt{\alpha_t(1-\alpha_{t-1})^2 + (1-\alpha_t)^2}\epsilon}_{\text{用一个新的高斯分布代替}} \\ &= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\epsilon \\ &= \sqrt{\prod_{i=1}^t \alpha_i}x_0 + \sqrt{1-\prod_{i=1}^t \alpha_i}\epsilon \\ &= \sqrt{\overline{\alpha_t}}x_0 + \sqrt{1-\overline{\alpha_t}}\epsilon, \overline{\alpha_t} = \prod_{i=1}^t \alpha_i, \epsilon \sim \mathcal{N}(0, I) \end{aligned}$$

我们发现只要设置了超参数  $\alpha_{0:T}$  的值，这个前向计算过程是可以直接解析（使用公式）计算的，没有未知参数，不需要用一个模型学习这个过程。

## 逆向过程

逆向过程是从右到左的解码过程，从随机高斯噪声开始，逐步解码为一个有意义的数。按照逆向过程对联合概率  $p(x_{0:T})$  进行分解  $p(x_{0:T}) = p(x_T)\prod_{t=T-1}^0 p(x_t|x_{t+1})$

我们可以知道  $p(x_T)$  的概率密度，是一个标准高斯分布，这是前向过程的目标。但是  $p_{\theta}(x_t|x_{t+1})$  是难以计算的。

$$p_{\theta}(x_t|x_{t+1}) = \frac{p(x_{t+1}|x_t)p(x_t)}{\int p(x_{t+1}|x_t)p(x_t)dx_t}$$

这种情况下要对所有可能的  $x_t$  进行积分，显然是不可能做到的。所以我们可以用一个模型去拟合  $p_{\theta}(x_t|x_{t+1})$  的，从而生成一张真实图片。

## 目标函数(ELBO)

同前面的VAE的数学推导，我们可以用极大似然估计来极大化真实图片概率 $p(x_0)$ （边际分布）。 $p(x_0) = \int p(x_{0:T}) dx_{1:T}$  显然无法直接通过这个式子求出 $p(x_0)$ ，存在隐变量无法直接积分，下面来推导**ELBO**。

$$\begin{aligned} \ln p(x_0) &= \ln \int p(x_{0:T}) dx_{1:T} \\ &= \ln \int \frac{p(x_{0:T}) q(x_{1:T}|x_0)}{q(x_{1:T}|x_0)} dx_{1:T} \\ &= \mathbb{E}_{q(x_{1:T}|x_0)} \left[ \ln \frac{p(x_{0:T})}{q(x_{1:T}|x_0)} \right] \\ &= \mathbb{E}_{q(x_{1:T}|x_0)} \left[ \ln p(x_T) \prod_{t=1}^T p_{\theta}(x_t|x_{t-1}) \right] \\ &= \mathbb{E}_{q(x_{1:T}|x_0)} \left[ \ln p_{\theta}(x_0|x_1) \prod_{t=1}^{T-1} p_{\theta}(x_t|x_{t+1}) \right] \\ &= \mathbb{E}_{q(x_{1:T}|x_0)} \left[ \ln p_{\theta}(x_0|x_1) + \sum_{t=1}^{T-1} \ln p_{\theta}(x_t|x_{t+1}) \right] \\ &= \mathbb{E}_{q(x_{1:T}|x_0)} \left[ \ln p_{\theta}(x_0|x_1) + \sum_{t=1}^{T-1} \mathbb{E}_{q(x_{t-1}, x_t, x_{t+1}|x_0)} \left[ \ln \frac{p_{\theta}(x_t|x_{t+1})}{q(x_t|x_{t-1})} \right] \right] \\ &= \underbrace{\mathbb{E}_{q(x_{1:T}|x_0)} [\ln p_{\theta}(x_0|x_1)]}_{\text{重建项}} - \underbrace{\mathbb{E}_{q(x_{T-1}, x_T|x_0)} [\ln \frac{p_{\theta}(x_T)}{q(x_T|x_{T-1})}]}_{\text{先验匹配项}} - \underbrace{\sum_{t=1}^{T-1} \mathbb{E}_{q(x_{t-1}, x_t, x_{t+1}|x_0)} [\ln \frac{p_{\theta}(x_t|x_{t+1})}{q(x_t|x_{t-1})}]}_{\text{一致项}} \end{aligned}$$

我们接着来说明倒数第三个等式如何推成倒数第二个等式：（本质是对**无关变量**进行边缘化）

核心公式：

$$\begin{aligned} q(x_{T-1}|x_0) &= \int q(x_1|x_0) q(x_2|x_1) \dots q(x_{T-1}|x_{T-2}) dx_{1:T-2} \\ &= \int q(x_1|x_0) q(x_2|x_1, x_0) \dots q(x_{T-1}|x_{T-2}, x_{T-3}) \dots, x_0) dx_{1:T-2} \\ &= \int \frac{q(x_{0:T})}{q(x_0)} dx_{1:T-2} \\ &= q(x_{T-1}|x_0) \end{aligned}$$

$$\begin{aligned} \mathbb{E}_{q(x_{1:T}|x_0)} [\ln p_{\theta}(x_0|x_1)] &= \int \ln p_{\theta}(x_0|x_1) q(x_{1:T}|x_0) dx_{1:T} \\ &= \int \ln p_{\theta}(x_0|x_1) q(x_1|x_0) \underbrace{\int \prod_{t=2}^{T-1} q(x_t|x_{t-1}) dx_{2:T}}_{1(\text{积分归一性})} dx_1 \\ &= \mathbb{E}_{q(x_1|x_0)} [\ln p_{\theta}(x_0|x_1)] \end{aligned}$$

$$\begin{aligned} \mathbb{E}_{q(x_{1:T}|x_0)} [\ln \frac{p(x_T)}{q(x_T|x_{T-1})}] &= \int \ln \frac{p(x_T)}{q(x_T|x_{T-1})} q(x_{1:T}|x_0) dx_{1:T} \\ &= \int \ln \frac{p(x_T)}{q(x_T|x_{T-1})} q(x_T|x_{T-1}) dx_{T-1, T} \end{aligned}$$

$$\underbrace{\int \prod_{t=1}^{T-1} q(x_t|x_{t-1}) dx_{1:T-2}}_{\text{积分归一性: } q(x_{T-1}|x_0)} \backslash \\ \&= \int \ln \left( \frac{p(x_T)}{q(x_T|x_{T-1})} \right) q(x_T|x_{T-1}) q(x_{T-1}|x_0) dx_{T-1,T} \backslash \&= \\ \int \ln \left( \frac{p(x_T)}{q(x_T|x_{T-1})} \right) q(x_{T-1}, x_T|x_0) dx_{T-1,T} \backslash \&= \mathbb{E} \\ \{q(x_{T-1}, x_T|x_0) \ln \left( \frac{p(x_T)}{q(x_T|x_{T-1})} \right)\} \end{align} \quad \quad \quad$$

$$\begin{aligned} \mathbb{E} \{q(x_{1:T}|x_0) [\sum_{t=1}^{T-1} \ln \left( \frac{p_{\theta}(x_t|x_{t+1})}{q(x_t|x_{t-1})} \right)] \} \&= \sum_{t=1}^{T-1} \int \ln \left( \frac{p_{\theta}(x_t|x_{t+1})}{q(x_t|x_{t-1})} \right) q(x_{1:T}|x_0) d_{x_{1:T}} \backslash \&= \sum_{t=1}^{T-1} \int \ln \left( \frac{p_{\theta}(x_t|x_{t+1})}{q(x_t|x_{t-1})} \right) q(x_t|x_{t-1}) \prod_{k=1}^T q(x_{k-1}|x_k) dx_{1:T} \&= \sum_{t=1}^{T-1} \int \ln \left( \frac{p_{\theta}(x_t|x_{t+1})}{q(x_t|x_{t-1})} \right) q(x_t|x_{t-1}) q(x_{t+1}|x_t) d_{x_{t-1}, x_t, x_{t+1}} \int \prod_1^t \end{aligned}$$

接着来看最后ELBO的三个式子：

- $\mathbb{E} \{q(x_1|x_0) \ln [p_{\theta}(x_0|x_1)]\}$ ：重建项，和原始的VAE的第一项相同，从第一步的隐变量 $x_1$ 重建回原来的数据 $x_0$
- $\mathbb{E} \{q(x_{T-1}, x_T|x_0) \ln \left( \frac{p(x_T)}{q(x_T|x_{T-1})} \right)\}$ ：先验匹配项，这一项没用学习参数，当 $T$ 足够大的时候可以认为这一项是0
- $\sum_{t=1}^{T-1} \mathbb{E} \{q(x_{t-1}, x_t, x_{t+1}|x_0) [\ln \left( \frac{p_{\theta}(x_t|x_{t+1})}{q(x_t|x_{t-1})} \right)]\}$ ：KL散度量，一致项，这一项制约着在每一个时刻 $t$ ，解码器预测的内容和编码器生成的内容要一致

## 个人理解

- 初始扩散模型前向过程的线性高斯变换已经给出，前向过程进行到最后是一个**标准高斯分布**，但是也可以采取不同的**线性高斯变换**，只要加噪到最后是一个标准高斯分布即可，殊途同归
- 扩散模型前向过程的参数是人为定义的，没有需要学习的参数
- 反向生成的过程：扩散模型反向生成的每一步都是希望**能够直接生成原始图像**的，在后续数学推导可见得，有点囫圇吞枣的意味，这说明他的性能有上限，设计具有局限性，后面的DDPI改善了此问题
- 扩散模型相较VAE能够学习到更深层的特征，因为有 $T$ 层的加噪和 $T$ 层的还原，VAE只有一步重建
- 计算期望的时候由于无法直接积分，采用采样法（MCMC）进行均值计算

## Denoising Diffusion Probabilistic Models(去噪扩散概率模型 DDPM)

## 训练算法

### 学习如何预测噪声，而不是直接生成图像

1. 从真实图像数据分布  $q(x_0)$  中采样一张干净图像  $x_0$
2. 从均匀分布  $\text{Uniform}(\{1, \dots, T\})$  中随机抽取一个时间步  $t$
3. 从  $\mathcal{N}(0, I)$  采样出噪声  $\epsilon$
4. 根据  $\|\nabla_{\theta} \|\epsilon - \epsilon_{\theta}(\sqrt{\alpha_t})x_0 + \sqrt{1-\alpha_t}\epsilon\|_{\theta}^2$  作梯度下降
  - $\alpha_t$  与第几次去噪有关，次数多说明真正的图片  $x_0$  占比大
  - 让模型学会预测噪声，以便在反向采样时去噪

 image-20250511112847361

## 反向采样算法

### 实际生成新图像

1. 从正态分布中采样一个图像  $x_T$
2. 从正态分布中采样一个噪声  $z$
3. 使用公式进行迭代  $x_{t-1} = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\overline{\alpha_t}}} \epsilon_{\theta}(x_t, t)) + \sigma_t z$ 
  - $x_{t-1}$  为第  $t-1$  次的去噪结果
  - $\alpha_t, \overline{\alpha}_t$  与第几次去噪有关
  - $\epsilon_{\theta}(x_t, t)$  是一个噪声预测器

 image-20250511111842681