现代概率机器学习初步

August 11, 2024

1 引言

当下, 生成式人工智能 (AGI) 广受关注, 在文本、图像、视频生成等领域均有不凡表现。其中, 在图像生成、视频生成等任务中, 目前效果最佳的技术路线为扩散模型 (Diffusion Model,DM), 例如 OpenAI 的 Sora 模型。尽管人工智能一向被大多数人视作不可解释的"黑箱", 但具体到扩散模型, 却有着极其清晰、深刻的数学本质: 可以说在数学视角下, 其本质上就是一个统计模型。本文尝试基于概率统计理论, 以严格的数学语言, 对扩散模型原理进行详细的推导与总结。值得注意的是, 当前扩散模型的理论研究, 更多已从初等概率论视角演变到随机分析视角, 通过运用 ODE、SDE 等工具, 将模型本质抽象到更加高等的数学中。但由于本课程的核心内容不涉及随机分析, 故本文仅从初等概率论与统计视角出发, 以扩散模型的"开山之作"——Jonathan Ho 等于 2020 年发表的 **去噪扩散概率模型** (Denoising Diffusion Probabilistic Models,DDPM) [1] 及其延伸话题为讨论对象。

值得注意的是, 由于论文 [1] 没有以数学的思路来提出、整理命题, 并且较多结论没有给出证明 (这是因为论文作者认为证明较为简单, 不便占用过多篇幅), 因此本文不会简单地重复该论文的思路与结构, 而是尝试整理其前因后果, 对其中出现的结论加以数学化的整理与表达。

本文将先介绍机器学习中, **生成式模型**的普遍性概念, 再具体探讨**扩散模型**的数学内涵。

符号含义与其他说明 1. 对随机向量 $(X_1, X_2, ..., X_n)^T$, 为简洁起见, 将之记作 X 。

2. 对随机向量 $X = [x_1, x_2, ..., x_n]^T$, $dX := dx_1 dx_2 ... dx_n$ 。

3. $p(X)$ 与 $q(X)$ 表示随机变量 (向量) X 的 PDF 。

4. $p_{data}(X)$ 表示数据实际服从的分布,$p_{model}(X)$ 表示建模得到的分布。

5. 如无特殊说明, 本文的所有随机变量 (向量) 的分布均为连续分布, 且其 PDF 均连续.

2 生成式模型的基本概念

机器学习的两种建模方式分别为判别式建模 (discriminative modeling) 和生成式建模 (generative modeling), 扩散模型即属于后者。因此, 我们先来探讨生成式模型的基本概念。

2.1 生成式模型

我们先以图像生成为例加以说明。计算机将图像表示成许多像素, 如果将每个像素视作一个随机变量, 那么整张图片便可以看做一个随机向量 X. 根据 **数据流形的分布假设**, 有意义的图像 (即现实生活中能够出现的图像) 服从某种概率分布, 即 $X \sim P_{data}(X)$. 我们的目的是, 通过对现

实数据的统计方法建立一个统计模型 $P_{model}(X)$，来模拟 $P_{data}(X)$ ，这就引出了生成式模型的概念：

定义 1 (生成式模型) ：给定特征空间 R^d 上的数据分布 $P_{data}(X)$, 生成式模型是由参数空间 Θ 中参数 θ 定义的一个参数模型，使得模型分布 $P_{model}(X)$ 近似于 $P_{data}(X)$，并可以从 $P_{model}(X)$ 中采样，从而生成新的数据。

生成式模型的一个重要作用，在于模型分布如能足够近似真实数据分布，则通过从模型分布中采样，可以生成近似服从真实分布的数据，从而实现全新数据的获得。例如，在完成图像生成模型的训练后，我们便可以源源不断地获取心目中高质量的图片。

2.2 生成式模型的两大任务

根据上述概念, 我们可以看出生成式模型的两个任务: 1. 尽量精确地拟合真实分布 $P_{data}(X)$, 我们将此称作 **训练**; 2. 从模型分布中采样, 以便生成新的数据, 我们将此称作 **推理**。对训练而言, 要估计 $P_{data}(X)$ ，我们有很多方法，此处以最大似然估计为例。给定数据样本 $\{X_1, X_2, ..., X_n\}$, 训练所得参数为

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \sum_{i=1}^{n} \log p_{model}(X_i).$$

2.3 隐变量生成模型

在明确基本概念后, 我们来考虑如何具体参数化 $P_{model}(X)$. 当前, 包括扩散模型在内的主流生成模型，均为 **隐变量生成模型**。以下将对此加以解释：

隐变量 首先明确 **隐变量**的概念。隐变量（latent variable）是不能直接观测到的随机变量，它用于描述数据生成过程中的未知因素。举例而言，在统计“北京市市民是否支持五环以内限行”问题时，我们获得的数据只有“支持还是反对”的信息（这就是所谓“观测变量”），而这些调查对象的年龄、居住地等信息，对于我们来说不可知。然而，这些因素却深切影响了”是否支持”这一观测变量，我们称这些因素为“隐变量”。

然后定义“隐变量生成模型”：

定义 2(隐变量生成模型) 对于某个生成模型, 设随机向量 X 为观测变量, Z 为隐变量, 若 $p_Z(z), p_{X|Z}(x|z)$ 是参数模型，则称这一模型为隐变量生成模型。

注 1 通常，隐变量边缘分布 $p_Z(z)$ 是**已知的或人为设定的先验分布**，如标准正态分布。在这种情况下，模型中唯一未定参数是 $p_{X|Z}(x|z)$ 中的参数。于是，参数估计 (及模型训练) 的任务就转化成：根据样本，使用 MLE 方法，确定 $p_{X|Z}(x|z)$ 的参数。

我们来稍作解释。为何要引入隐变量? 这可以大大提升模型的表达能力，便于参数化, 有利于使用机器学习方法予以拟合。举一个具体例子：

例 1(变分自编码器的译码器模型, Decoder of VAE) 设隐变量 $Z \sim N(Z; 0, I)$, 特征变量 (如图像像素) 为 X, 定义 $p_{X|Z}(x|z) = N(x; \mu(z); \sigma^2(z)I)$, 其中 $\mu(z), \sigma(z)$ 均为 z 的函数。则 X 的模型分布为 $p_{model}(x) = p_Z(z)p_{X|Z}(x|z)$. (注：高维正态分布的刻画涉及协方差矩阵，此处是设其协方差阵为一个对角阵 $\sigma^2(z)I$. 下文若无特殊说明，所涉及的正态分布均为高维正态分布)

对于上例，我们用神经网络来拟合 $\mu(z),\sigma(z)$ 两个函数。假设 $\mu(z) = f(z;\theta_1,\theta_2,...,\theta_k)$ ，其中 θ_i 为参数，则给定数据样本 $\{x_1,x_2,...,x_n\}$ ，神经网络的训练目标是:

$$\theta_i^* = \underset{\theta_i}{\operatorname{argmax}}(\sum_{i=1}^{n} \log p_{model}(x_i)) \tag{1}$$

$$= \underset{\theta_i}{\operatorname{argmax}}(\sum_{i=1}^{n} \log \int p(x_i,z)dz) \tag{2}$$

$$= \underset{\theta_i}{\operatorname{argmax}}(\sum_{i=1}^{n} \log \int p_Z(z)p_{X|Z}(x_i|z)dz) \tag{3}$$

$$= \underset{\theta_i}{\operatorname{argmax}}(\sum_{i=1}^{n} \log \int N(z;0,I)N(x_i;\mu(z),\sigma^2(z)I)dz, \tag{4}$$

其中 $N(z;0,I)$ 表示将 z 代入标准正态分布 $N(0,I)$ PDF 后所得结果，$N(x_i;\mu(z),\sigma^2(z)I)$ 表示将 x_i 代入正态分布 $N(x;\mu(z),\sigma^2(z)I)$ 的 PDF 后所得结果。

显然，相较于直接假设 $p_{model}(X)$ 是正态分布 $N(\mu,\sigma^2 I)$, 隐变量模型的参数更多 (直接假设，则参数只有 μ, Σ 两个常数，而隐变量模型则可以将 $\mu(z)$,$\sigma^2(z)$ 视作函数，用神经网络的许多参数来拟合它们。我们有理由认为，参数越多，模型的表达能力越强 (也就是是能够更加逼近真实的数据分布 $P_{data}(X)$))。

2.4 隐变量模型的变分推断

隐变量生成模型的训练具体应怎么操作? 我们此处一直假定采用 MLE 方法。沿用上述例子，如何来具体求出

$$\theta_i^* = \underset{\theta_i}{\operatorname{argmax}}(\sum_{i=1}^{n} \log \int p_Z(z)p_{X|Z}(x_i|z)dz) \tag{5}$$

$$= \underset{\theta_i}{\operatorname{argmax}}(\sum_{i=1}^{n} \log p_X(x_i)), \tag{6}$$

是一个关键问题。显然，想要直接求导数做最大似然，就需要求出上式积分的一个解析解，这无疑非常困难。换言之，隐变量参数模型的训练，不能直接采用对数似然来做 MLE。为此，我们引入 **变分推断**的方法，以实现 MLE 的具体操作。

命题 1 (对数似然的变分下界) 记对数似然 $L(\theta) := \sum_{i=1}^{n} \log p_X(x_i)$，$q(z|x;\phi)$ 为另一个分布，其中 ϕ 是参数。则:

$$L(\theta) \geq \sum_{i=1}^{N} E_q[\log p(z,x_i;\theta) - \log q(z|x_i;\phi)] := \hat{L}(\theta,\phi), \tag{7}$$

且若 q 的函数形式与 ϕ 的具体取值均无任何限制，则对任意确定的 p 与 θ, 都存在一个 q 的函数形式与 ϕ 的取值，使得 $\hat{L}(\theta,\phi) = L(\theta)$. 证明见附录A .

根据此命题, 可知 $max_{\theta,\phi} L(\hat{\theta,\phi})) = max_\theta\ L(\theta)$, 从而 $\operatorname{argmax}_\theta(max_\phi L(\hat{\theta,\phi})) = \operatorname{argmax}_\theta(L(\theta))$, 因此对 $\hat{L}(\theta,\phi)$ 做 MLE 来估计 θ^* ，等价于对 $L(\theta)$ 做 MLE.

基于此，我们对 $q(z|x;\phi)$ 做一些预设（例如，确定 q 的函数形式，确定其参数个数），使得 $\hat{L}(\theta,\phi)$ 易于实际计算，那么我们便将难以实际操作的 $L(\theta)$ MLE 转换成方便计算的 $\hat{L}(\theta,\phi)$ MLE.

当然，由于为了便于计算，我们对 $q(z|x;\phi)$ 做了一些预设，这些限制可能导致上述的取等无法实现。但事实上，即使不能取等，二者的差异也相对较小。因此，这种方法是进行 MLE 的可行方式。

实际上，这就是所谓 **变分推断**的基本思想，扩散模型正是基于这一思路进行参数估计（也即所谓"训练"）的。下面，我们正式进入对扩散模型的讨论。

3 扩散模型：基于概率统计的数学本质

扩散模型是一个 **多隐变量**生成模型，在上文隐变量模型的基础上，引入了多个隐变量，从而实现了更优的表达能力。下面先介绍模型的参数化形式，随后在此基础上说明参数估计（即"模型训练"）的具体方法。

3.1 模型的参数化形式

定义 2 (去噪扩散概率模型) 设随机向量 $X_0,X_1,...,X_T\in R^d$, 其中 R^d 为特征空间，X_0 为观测变量（eg. 图像像素组成的随机向量），$X_1,...,X_T$ 为隐变量。定义隐变量 $X_i(i=1,2,..,T)$ 服从的真实分布为

$$q_{X_0}(x_0)=p_{data}(x_0), \tag{8}$$

$$q_{X_t|X_{t-1}}(x_t|x_{t-1})=N(x_t;\sqrt{1-\beta_t}x_{t-1},\beta_tI), \tag{9}$$

$$q_{X_t|X_{t-1},...,X_0}(x_t|x_{t-1},...,x_0)=q_{X_t|X_{t-1}}(x_t|x_{t-1})\quad ,t=1,2,..,T, \tag{10}$$

模型分布为

$$p_{X_T}(x_t)=N(x_T;0,I), \tag{11}$$

$$p_{X_{t-1}|X_t}(x_{t-1}|x_t)=N(x_{t-1};\mu(x_t,t),\sigma_t^2I), \tag{12}$$

$$p_{X_{t-1}|X_t,...,X_T}(x_{t-1}|x_t,...,x_T)=p_{X_{t-1}|X_t}(x_{t-1}|x_t)\qquad ,t=1,2,..,T, \tag{13}$$

其中 σ_t 是预设常数（是不可学习的定值），$\mu(x_t,t)$ 是关于 x_t 与 t 的函数，使用神经网络进行模拟。以这个神经网络中的参数作为待估计参数，则这个参数模型称作**去噪扩散概率模型** (Denoising Diffusion Probabilistic Model，DDPM)。

下面对此概念进行一些直观的解释。X_0 是数据特征变量，例如一张 256*256、由三原色组成的图像，则其中所有像素构成了 $X_0\in R^{256\times256\times3}$。隐变量 $X_1,...,X_T$ 是人为引入的变量，而既然是人为引入，那么就可以人为指定它们的分布形式，具体而言，$q_{X_t|X_{t-1}}(x_t|x_{t-1})=N(x_t;\sqrt{1-\beta_t}x_{t-1},\beta_tI)$ 这一条件分布，指明了 X_i 的构造方法：根据正态分布性质，联系上式，我们有

$$X_t=\sqrt{1-\beta_t}X_{t-1}+\sqrt{\beta_t}\epsilon,\ \ \epsilon\sim N(0,I). \tag{14}$$

换言之，X_t 其实就是对 X_{t-1} 与标准正态 ϵ 的线性组合。由于标准正态分布是不含有效信息的高斯噪声，因此获得 X_t 的这一过程，就相当于向 X_{t-1} 添加一些"噪声"。因此，这个过程被称作**"加噪过程"**（也称"前向过程"）。图1是 $X_0,X_1,...,X_T$ 的简单例子. 例子中 $T=6$ ，而事实上为

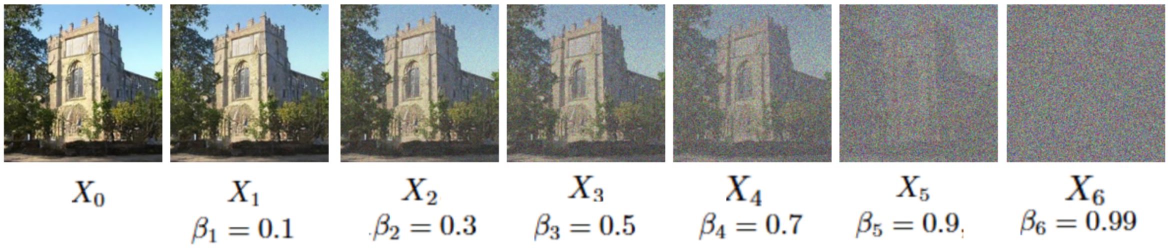

Figure 1: **隐变量分布的定义: 加噪过程**

提高模型表达能力，T 往往很大，如 $T = 1000$. 至于 10 式所规定的马尔可夫性，则其实也能反应在式 14 中，即: 隐变量的分布仅取决于它的直接前驱，与时间序列中的其他祖先无关。

类似地，我们定义的模型分布 $p_{X_{t-1}|X_t}(x_{t-1}|x_t) = N(x_{t-1}; \mu(x_t, t), \sigma_t^2 I)$ $(t = 1, 2, .., T)$ 可以理解成从一个标准正态分布出发不断迭代，用许多正态分布的 "复合" 来近似真实的数据分布。具体而言:

$$
\begin{aligned}
p_{model}(x_0) &= p_{X_0}(x_0) && (15)\\
&= \int p(x_0, x_1, ..., x_T) dx_1...dx_T && (16)\\
&= \int p_{X_T}(x_T) \prod_{t=1}^{T} p_{X_{t-1}|X_t}(x_{t-1}|x_t) dx_1...dx_T, && (17)
\end{aligned}
$$

其中 16 式到 17 式的等号利用了 13 式规定的马尔可夫性。

换言之，只要我们能够确定 $p_{X_{t-1}|X_t}(x_{t-1}|x_t)$, 我们便在理论上获得了模型分布 $p_{model}(x_0)$. 当然，即使确定了这些参数，由于计算的复杂性，我们还是不可能直接积分来求 $p_{model}(x_0)$ ，具体如何根据这些隐变量条件分布来实现从 $p_{model}(x_0)$ 中采样，我们将在后文略作讨论。

下面，我们从 "生成模型的两大任务" (见 2.2)，即 **训练**与 **推理**两方面展开。

3.2 扩散模型的训练（模型参数估计）

注 在实际应用神经网络进行训练时, 为了便于减小误差、更易拟合, 原论文并未直接拟合 $\mu(x_t, t)$ ，而是采用了一个等价形式。然而由于其数学本质没有区别，更多是工程上的技巧，因此为了使表述简洁而不冗杂，本文仍直接用 $\mu(x_t, t)$ 的拟合来推导原理。

3.2.1 对数似然及其变分下界

我们的目标是通过 **最大似然估计**来做参数估计。但是，沿用 2.4 提出的观点，由于直接计算边际分布需要积分，但此积分在实际操作中几乎不可积，因此我们不能直接运用 17 式，求出 $p(X_0)$ 之后做 MLE ，而该使用 **变分推断**的方法。具体而言，设训练集为 $\{x_{01}, x_{02}, ..., x_{0N}\}$, 隐变量 $X_{ti}(t = 1, .., T)$ 是以 X_{0i} 为基础，根据加噪过程构造的（注，针对训练集的每个 X_{0i}, 我们构造许多 X_{ti}，以增强训练效果）。用上述 $X_{1i}, ..., X_{Ti}$ 代入式 7 中的隐变量 z ，可得:

$$
\sum_{i=1}^{N} \log p_{X_0}(x_{0i}) \geq \sum_{i=1}^{N} E_{q(X_1,...,X_T)}[\log p(x_{1i}, x_{2i}, ..., x_{Ti}, x_{0i}; \theta) - \log q(x_{1i}, x_{2i}, ..., x_{Ti}|x_{0i})], \quad (18)
$$

这里 RHS 是对 $X_1, ..., X_T$ 求期望。当 N 极大时，由大数定律，该式近似为

$$E_{q(X_0)} \log p_{X_0}(x_0) \geq E_{q(X_0,X_1,...,X_T)}[\log p(x_1, x_2, ..., x_T, x_0; \theta) - \log q(x_1, x_2, ..., x_T|x_0)], \tag{19}$$

这里 LHS 是对 X_0 求期望，RHS 是对 $X_0, X_1, ..., X_T$ 求期望。由于实际训练的 N 非常大，因此下文默认采用此式。

于是，MLE 中的

$$\theta^* = \underset{\theta}{\operatorname{argmax}}\, E_{q_{X_0}(X_0)} \log p_{X_0}(x_{0i}), \tag{20}$$

可以根据 2.4 而近似为

$$\theta^* = \underset{\theta}{\operatorname{argmax}}\, E_{q(X_0,X_1,...,X_T)}[\log p(x_1, x_2, ..., x_T, x_0; \theta) - \log q(x_1, x_2, ..., x_T|x_0)], \tag{21}$$

根据机器学习的优化习惯，我们希望最小化目标函数，故此对上式取反：

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\, E_{q(X_0,X_1,...,X_T)}[\log q(x_1, x_2, ..., x_T|x_0) - \log p(x_1, x_2, ..., x_T, x_0; \theta)]. \tag{22}$$

根据条件概率定义，联系 10 13 两式，我们可以把 22 式展开为

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\, E_{q(X_0,X_1,...,X_T)}[-\log p_{X_T}(X_T) - \sum_{t\geq 1} \log \frac{p_{X_{t-1}|X_t}(x_{t-1}|x_t)}{q_{X_t|X_{t-1}}(x_t|x_{t-1})}] \tag{23}$$

$$:= \underset{\theta}{\operatorname{argmin}}\, L. \tag{24}$$

下面来最小化 L 。

3.2.2 参数估计的具体目标推导

命题 2

$$L = E_q[KL(\ q(x_T|x_0) \| p(x_T)\ ) + \sum_{t>1} KL(\ q(x_{t-1}|x_t, x_0) \| p(x_{t-1}|x_t)\ ) - \log p(x_0|x_1)], \tag{25}$$

其中

$$KL(p(x) \| q(x)) := E_{p(x)}[\log \frac{p(x)}{q(x)}], \tag{26}$$

称为两个分布之间的 **KL 散度**。证明见附录 B(为简洁起见，上述命题将形如 $p_{X|Y}(x|y)$ 的条件概率符号记作 $p(x|y)$ 省略了下标).

考察 25 式中的三个加项，将之分别记作

$$L_T := E_q[KL(\ q(x_T|x_0) \| p(x_T)\ )], \tag{27}$$

$$L_{t-1} := E_q[KL(\ q(x_{t-1}|x_t, x_0) \| p(x_{t-1}|x_t)\ )], \tag{28}$$

$$L_0 := E_q[-\log p(x_0|x_1)]], \tag{29}$$

则

$$L = L_T + \sum_{t>1} L_{t-1} + L_0, \tag{30}$$

我们对这三项逐一加以探讨。

(i). 第一部分: 考察 L_T
考虑如下命题:

命题 3 加噪过程中,

$$q_{X_T|X_0}(x_T|x_0) = N(x_T; \sqrt{\alpha_T}\ x_0\ , (1-\alpha_T)I), \tag{31}$$

其中 $\alpha_T := \prod_{s=1}^{T}(1-\beta_s)$。证明见附录 C.

根据 **命题 3** , 我们发现 $q_{X_T|X_0}(x_T|x_0)$ 是一个不含可学习参数的分布（如上文所述，β_t 预先固定），而 $p_{X_T}(x_T)$ 也是人为预设的 $N(0, I)$ ，因此 L_T 没有可以推断的参数，是一个常数。

(ii). 第二部分: 考察 L_0
由于相较于 $T = 1000$ 这样庞大的数字，这一项对整体优化影响很小，且在实践中发现优化此项带来的性能提升极少，因此我们暂不考虑 L_0 的优化 (事实上 DDPM 的原始论文也未曾考察此项)。

(iii). 第三部分: 考察 L_{t-1}
因此，我们只需最小化 $\sum_{t>1} L_t$，我们这里尝试对每一个 $t \in \{2, 3, 4, ..., T\}$ ，都最小化 L_{t-1} 。事实上，这样做会自然地引发一个问题，即：这些 L_{t-1} 是否能够同时取最小值？假若不能，那么分别取最小，就不能确保整体之和最小。所幸，由于我们的参数是 **一个拟合** $\mu(x, t)$ **函数的神经网络**中的参数，而神经网络只要足够深，拟合能力便相当之强，因此足以实拟合对每个 t 都取 $minL_{t-1}$ 的 $\mu(x, t)$.
先考虑如下两个命题:

命题 4(离散形式的布朗桥) 加噪过程中,

$$q(x_{t-1}|x_t, x_0) = N(x_{t-1}; \hat{\mu_t}(x_0, x_t), \hat{\beta}_t I), \tag{32}$$

其中

$$\hat{\mu_t}(x_0, x_t) := \frac{\sqrt{\alpha_{t-1}}\beta_t}{1-\alpha_t}x_0 + \frac{\sqrt{1-\beta_t}(1-\alpha_{t-1})}{1-\alpha_t}x_t, \tag{33}$$

$$\hat{\beta}_t := \frac{1-\alpha_{t-1}}{1-\alpha_t}\beta_t. \tag{34}$$

证明见附录 D.

命题 5(正态分布之间的 KL 散度公式) 设随机变量 $X \in R^d$, 概率密度函数

$$p_1(x) = N(\mu_1, \sigma_1^2 I), p_2(x) = N(\mu_2, \sigma_2^2 I),$$

则这两个正态分布之间的 KL 散度为

$$KL(p_1(x)\|p_2(x)) = \frac{1}{2}\log(\frac{\sigma_1^2}{\sigma_2^2})^d + \frac{\sigma_1^2 + \|\mu_1-\mu_2\|^2}{2\sigma_2^2} - \frac{d}{2}. \tag{35}$$

这里仅给出了正态分布的协方差阵是对角阵的情况，不过对我们的话题已经足够。证明见附录 E.

根据 **命题 4**、**命题 5** ，我们可以将 L_{t-1} 化简：

$$
\begin{aligned}
L_{t-1} &= E_q[KL(\ q(x_{t-1}|x_t,x_0)\|p(x_{t-1}|x_t)\ )] &(36)\\
&= E_q[KL(\ N(x_{t-1};\hat{\mu}_t(x_0,x_t),\hat{\beta}_t I)\ \|\ N(x_{t-1};\mu(x_t,t),\sigma_t^2 I\ )] &(37)\\
&= E_q[\frac{1}{2}\log(\frac{\hat{\beta}_t}{\sigma_t^2})^d + \frac{\hat{\beta}_t + \|\hat{\mu}_t(x_0,x_t)-\mu(x_t,t)\|^2}{2\sigma_t^2} - \frac{d}{2}]. &(38)
\end{aligned}
$$

38 式中，$\hat{\beta}_t$ 仅取决于 x_0, x_t ，而这两个变量是训练数据，因此是常数；σ_t 是预设常数；d 是特征空间维度，也是常数。因此，想要最小化 L_{t-1} ，可优化之处只有 $\|\hat{\mu}_t(x_0,x_t)-\mu(x_t,t)\|^2$. 因此:

$$
\begin{aligned}
\theta_{t-1}^* &= argmin\ L_{t-1} &(39)\\
&= argmin\ E_q[\frac{1}{2}\log(\frac{\hat{\beta}_t}{\sigma_t^2})^d + \frac{\hat{\beta}_t + \|\hat{\mu}_t(x_0,x_t)-\mu(x_t,t)\|^2}{2\sigma_t^2} - \frac{d}{2}] &(40)\\
&= argmin\ E_q[\|\hat{\mu}_t(x_0,x_t)-\mu(x_t,t)\|^2]. &(41)
\end{aligned}
$$

至此，我们终于得到了参数 θ 的表达式（亦即参数估计方法）：

$$
\theta_{t-1}^* = argmin\ E_q[\|\hat{\mu}_t(x_0,x_t)-\mu(x_t,t)\|^2]. \tag{42}
$$

有了这个解析式，我们就可以采用梯度下降等方法不断优化我们的神经网络 (即 $\mu(x_t,t)$)，从而完成训练。

小结 回顾上述推导过程，我们从最大似然估计出发，由于对数似然包含复杂积分、不能直接计算，因此我们使用变分推断，通过引入对数似然的变分下界，将 MLE 的目标转换成 $argminL$。在此基础上，我们通过命题 2 (式 25) , 将 L 拆分成 $L_T, \sum_{t>1} L_{t-1}, L_0$ 三部分，证明了 L_T 是常数，于是在 L_0 的影响极小、可以忽略的前提下，我们的 MLE 等价于最小化 $\sum_{t>1} L_{t-1}$。由于神经网络极强的拟合能力，这一目标可以通过令所有 L_t 同时取最小实现。通过证明 L_t 所涉及的两个分布均为正态分布，我们就可以通过正态分布之间的 KL 散度公式来得到 $minL_t$ 的等价的、可操作的形式。由此，我们最终找到了一个可操作的参数估计方法。

3.3 扩散模型的推理（从分布中采样）

既已确定参数，我们下面就可以利用这个模型来进行生成。正如上文 (3.1) 所述，在确定 $p_{X_{t-1}|X_t}(x_{t-1}|x_t)$ 后，我们不能直接通过 17 式积分来获得 $p_{model}(x_0)$ 。但是我们之所以要训练这样一个模型，正是为了能够从 $p_{model}(x_0)$ 中采样，以便获得新数据。为此，我们采用如下的采样方法:

算法 1（DDPM Sampler） :

1. 从 $N(0,I)$ 采样 X_T;

2. 遍历 $t = T, ..., 1$, 进行如下操作:

   (i). 从 $N(0,I)$ 采样 z_t;

   (ii). 计算 $X_{t-1} = \mu(X_t,t) + \sigma_t z_t$.

3. 获得 $X_0 \sim P_{model}(X_0)$.

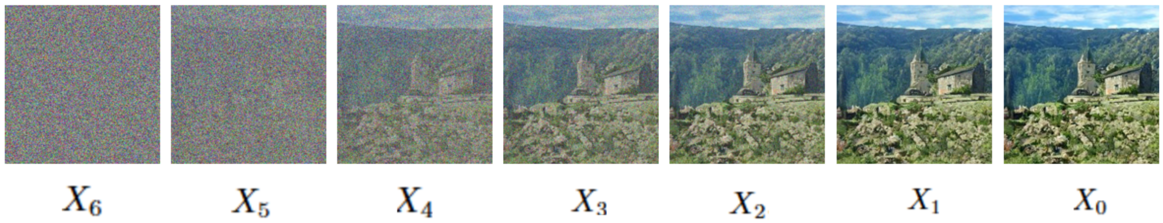

Figure 2: **模型推理: 去噪过程**

这其实就是利用正态分布的性质把 $p_{X_{t-1}|X_t}(x_{t-1}|x_t) = N(x_{t-1};\mu(x_t,t),\sigma_t^2 I)$ 重写一遍，但好处是这成为了一个可以实际操作的算法。这个算法的执行过程，由于是从高斯噪声出发，逐渐获得一个清晰的图片，因此被称作 **"去噪过程"**（或**反向过程**）. 图 2 以 $T=6$ 作为例子。值得注意的是，实际应用中 T 极大（eg.1000），那么就需要极多的迭代步数，这无疑导致了模型生成速度的下降。为解决这一问题，各种采样加速算法不断提出，如 DDIM [3]，DPM-Solver [2] 等等。篇幅所限，不加陈述。

4 未尽话题与结语

尽管本文对扩散模型的基本概念进行了一些讨论，尝试推导了一些公式，但本文距扩散模型的全貌仍相差很多，有太多的未尽话题值得思考：例如，上文的探讨有意回避了 **"条件生成"** 的概念，模型所生成的内容可能是任何一种物体，如何控制模型生成我们想要的内容？（例如，如何用文字控制模型的生成内容，eg. 输入 "一条狗追逐一只猫" 的文字，要求模型成对应内容的图像）这种 **条件生成**的概念，正是颇具热度的 **文生图**、**文生视频**等模型必不可少的基础。又如，上文推导基础原理时，曾将加噪过程中的方差强行固定，那么是否存在数学上的最优解，能够 **解析地确定最优方差**？再如，如上文所述，采样阶段的近一千步迭代将大大降低采样速率，这势必导致应用上的极大不便，如何实现既 **加速采样**，又保证生成质量，这些改进的背后又有着怎样的数学本质？这些问题引出的杰出工作，都曾大放异彩。至于如何将模型背后的数学本质提高到更加高等的领域、如何将这些离散的加噪、去噪过程转换成连续过程，这一问题引出了 Score Matching SDE 及与之等价的 ODE ，这些视角甫一提出，便成为理论研究领域不可或缺的出发点。时至今日，扩散模型的理论日趋完备，如何设计神经网络的具体结构、如何提高模型的生成质量，则成为工业界最为关心的话题。限于篇幅，本文仅探讨了扩散模型的 "开山之作"，未曾涉及这些精彩纷呈的后续内容。然而管中窥豹，可见一斑，通过对其数学本质的探讨，我们可以看到，当代人工智能并非毫无理论的 "黑盒"，其背后的统计机器学习原理之深刻精彩，诚为动人心魄。

最后，展示一些由扩散模型生成的图像：图 3；图 4 。

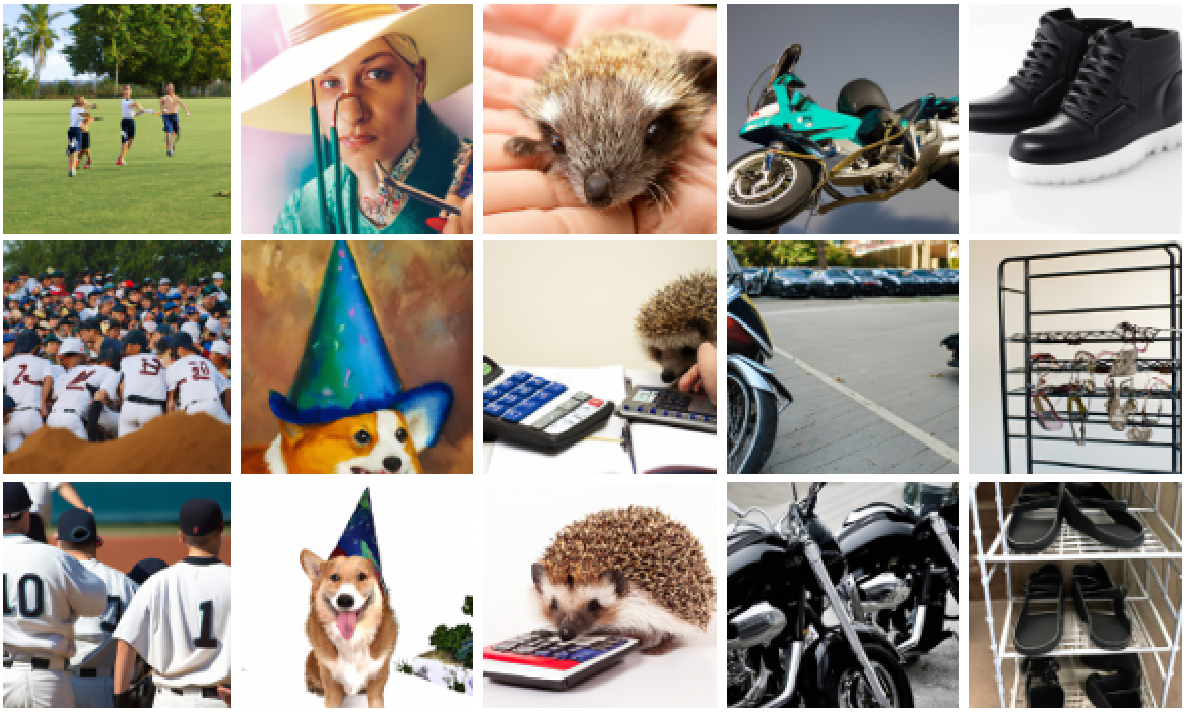

Figure 3: **扩散模型所生成图像的欣赏**

Figure 4: **扩散模型所生成图像的欣赏**

A 命题 1 证明

先证一条引理:

引理 1(信息不等式) 对任意两分布 $P(X), Q(X)$, 设其 PDF 分别是 $p(x), q(x)$, 均为连续函数, 则有

$$E_{q(x)}[\log \frac{q(x)}{p(x)}] \geq 0, \tag{43}$$

等号成立当且仅当 $\forall x, p(x) = q(x)$.

证明 1 以下给出 **引理 1** 的（不严格）证明:

$$E_{q(x)}[\log \frac{q(x)}{p(x)}] = -E_{q(x)}[\log \frac{p(x)}{q(x)}], \tag{44}$$

由积分形式的 Jensen 不等式,

$$-E_{q(x)}[\log \frac{p(x)}{q(x)}] \geq -\log E_{q(x)}[\frac{p(x)}{q(x)}] = -\log(\int p(x)dx) = 0. \tag{45}$$

由此得证 (事实上，仅使用 Jensen 不等式并不严格，尤其是取等条件难以说清。但严格证明涉及泛函分析知识，此处不做展开)。

下面证明 **命题 1**:

证明 2 以 z 为变量,x_i 为具体数值，由式 43 有

$$E_q[\log \frac{q_{Z|X}(z|x_i)}{p_{Z|X}(z|x_i)}] \geq 0, \tag{46}$$

由条件概率定义有

$$\begin{aligned}
E_q[\log \frac{q_{Z|X}(z|x_i)}{p_{Z|X}(z|x_i)}] &= E_q[\log \frac{q_{Z|X}(z|x_i)p_X(x_i)}{p(z,x_i)}] && (47)\\
&= E_q[\log \frac{q_{Z|X}(z|x_i)}{p(z,x_i)} + \log p_X(x_i)] && (48)\\
&= \log p_X(x_i) + E_q[\log q_{Z|X}(z|x_i) - \log p(z,x_i)] && (49)\\
&\geq 0, && (50)
\end{aligned}$$

即

$$\log p_X(x_i) \geq E_q[\log p(z,x_i) - \log q_{Z|X}(z|x_i)]. \tag{51}$$

于是证毕。

B 命题 2 证明

$$
\begin{aligned}
L &= E_q[-\log p(x_T) - \sum_{t\geq 1} \log \frac{p(x_{t-1}|x_t)}{q(x_t|x_{t-1})}] && (52)\\
&= E_q[-\log p(x_T) - \sum_{t>1} \log \frac{p(x_{t-1}|x_t)}{q(x_t|x_{t-1})} - \log \frac{p(x_0|x_1)}{q(x_1|x_0)}] && (53)\\
&= E_q[-\log p(x_T) - \sum_{t>1} \log \frac{p(x_{t-1}|x_t)}{q(x_{t-1}|x_t,x_0)} \cdot \frac{q(x_{t-1}|x_0)}{p(x_t|x_0)} - \log \frac{p(x_0|x_1)}{q(x_1|x_0)}] && (54)\\
&= E_q[-\log \frac{p(x_T)}{q(x_T|x_0)} - \sum_{t>1} \log \frac{p(x_{t-1}|x_t)}{q(x_{t-1}|x_t,x_0)} - \log p(x_0|x_1)] && (55)\\
&= E_q[KL(q(x_T|x_0)\|p(x_T)) + \sum_{t>1} KL(q(x_{t-1}|x_t,x_0)\|p(x_{t-1}|x_t)) - \log p(x_0|x_1)]. && (56)
\end{aligned}
$$

此处为了方便而将形如 $p_{X|Y}(x|y)$ 的符号记作 $p(x|y)$.

C 命题 3 证明

将命题加强为

$$q_{X_t|X_0}(x_t|x_0) = N(x_t; \sqrt{\alpha_t}\ x_0\ , (1-\alpha_t)I), \tag{57}$$

对 t 归纳，
当 $t=1$ 时，由 9 式，显然；
假设 $t=k$ 时成立，则 $t=k+1$ 时，

$$X_{k+1} = \sqrt{1-\beta_{k+1}}X_k + \sqrt{\beta_{k+1}}\epsilon_1,\ \epsilon_1 \sim N(0,I). \tag{58}$$

由归纳假设，

$$X_k = \sqrt{\alpha_k}X_0 + \sqrt{(1-\alpha_k)}\epsilon_0,\ \epsilon_0 \sim N(0,I). \tag{59}$$

因此:

$$
\begin{aligned}
X_{k+1} &= \sqrt{(1-\beta_{k+1})\alpha_k}X_0 + \sqrt{(1-\beta_{k+1})(1-\alpha_k)}\epsilon_0 + \sqrt{\beta_{k+1}}\epsilon_1,\ \epsilon_0,\epsilon_1 \sim N(0,I) && (60)\\
&= \sqrt{(1-\beta_{k+1})\alpha_k}X_0 + \sqrt{((1-\beta_{k+1})(1-\alpha_k))^2 + \beta_{k+1}}\epsilon,\ \epsilon \sim N(0,I) && (61)
\end{aligned}
$$

联系 α_{k+1} 定义整理即得:

$$X_{k+1} = \sqrt{\alpha_{k+1}}X_0 + \sqrt{1-\alpha_{k+1}}\epsilon,\ \epsilon \sim N(0,I), \tag{62}$$

即:

$$X_{k+1} \sim N(X_{k+1};\ \sqrt{\alpha_{k+1}}X_0, (1-\alpha_{k+1})I). \tag{63}$$

综上得证。

D 命题 4 证明

由条件概率定义:

$$q(x_{t-1}|x_t, x_0) = \frac{q(x_t, x_{t-1}, x_0)}{q(x_t, x_0)} \tag{64}$$

将分子分母展开后，由 10 式的马尔可夫性化简分子，:

$$\begin{aligned}
\frac{q(x_t, x_{t-1}, x_0)}{q(x_t, x_0)} &= \frac{q(x_t|x_{t-1}, x_0)q(x_{t-1}|x_0)q(x_0)}{q(x_t|x_0)q(x_0)} && (65)\\
&= \frac{q(x_t|x_{t-1})q(x_{t-1}|x_0)q(x_0)}{q(x_t|x_0)q(x_0)} && (66)\\
&= \frac{q(x_t|x_{t-1})q(x_{t-1}|x_0)}{q(x_t|x_0)}. && (67)
\end{aligned}$$

由加噪过程定义式 9 可得 $q(x_t|x_{t-1})$ ，由**命题 2** 25 式可得 $q(x_{t-1}|x_0)$ 和 $q(x_t|x_0)$，这些均为正态分布的 PDF，将解析式代入 67 整理即得。

E 命题 5 证明

协方差阵为对角阵的正态分布可由一元正态分布迁移而来，因此我们仅证明 $d = 1$ 的情况，即两个一元正态分布之间的 KL 散度公式:

$$\begin{aligned}
&KL(p_1(x)\|p_2(x))\\
&\overset{\text{根据定义}}{=} \int p_1(x) log\frac{p_1(x)}{p_2(x)}dx\\
&= \int p_1(x)(logp_1(x)dx - logp_2(x))dx = \int p_1(x) * (log\frac{1}{\sqrt{2\pi\sigma_1^2}}e^{\frac{(x-\mu_1)^2}{2\sigma_1^2}} - log\frac{1}{\sqrt{2\pi\sigma_2^2}}e^{\frac{(x-\mu_2)^2}{2\sigma_2^2}})dx\\
&= \int p_1(x) * (-\frac{1}{2}log2\pi - log\sigma_1 - \frac{(x-\mu_1)^2}{2\sigma_1^2} + \frac{1}{2}log2\pi + log\sigma_2 + \frac{(x-\mu_2)^2}{2\sigma_2^2})dx\\
&= \int p_1(x)(log\frac{\sigma_2}{\sigma_1} + [\frac{(x-\mu_2)^2}{2\sigma_2^2} - \frac{(x-\mu_1)^2}{2\sigma_1^2}])dx\\
&= \int (log\frac{\sigma_2}{\sigma_1})p_1(x)dx + \int (\frac{(x-\mu_2)^2}{2\sigma_2^2})p_1(x)dx - \int (\frac{(x-\mu_1)^2}{2\sigma_1^2})p_1(x)dx\\
&= log\frac{\sigma_2}{\sigma_1} + \frac{1}{2\sigma_2^2}\int ((x-\mu_2)^2)p_1(x)dx - \frac{1}{2\sigma_1^2}\int ((x-\mu_1)^2)p_1(x)dx\\
&\\
&= log\frac{\sigma_2}{\sigma_1} + \frac{1}{2\sigma_2^2}\int ((x-\mu_2)^2)p_1(x)dx - \frac{1}{2}\\
&= log\frac{\sigma_2}{\sigma_1} + \frac{1}{2\sigma_2^2}\int ((x-\mu_1+\mu_1-\mu_2)^2)p_1(x)dx - \frac{1}{2}\\
&= log\frac{\sigma_2}{\sigma_1} + \frac{1}{2\sigma_2^2}[\int (x-\mu_1)^2p_1(x)dx + \int (\mu_1-\mu_2)^2p_1(x)dx + 2\int (x-\mu_1)(\mu_1-\mu_2)]p_1(x)dx - \frac{1}{2}\\
&= log\frac{\sigma_2}{\sigma_1} + \frac{1}{2\sigma_2^2}[\int (x-\mu_1)^2p_1(x)dx + (\mu_1-\mu_2)^2] - \frac{1}{2}\\
&\\
&= log\frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_1-\mu_2)^2}{2\sigma_2^2} - \frac{1}{2}
\end{aligned}$$

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising Diffusion Probabilistic Models". In: *arXiv preprint arxiv:2006.11239* (2020).

[2] Cheng Lu et al. "DPM-Solver: A Fast ODE Solver for Diffusion Probabilistic Model Sampling in Around 10 Steps". In: *arXiv preprint arXiv:2206.00927* (2022).

[3] Jiaming Song, Chenlin Meng, and Stefano Ermon. "Denoising Diffusion Implicit Models". In: *arXiv:2010.02502* (Oct. 2020). URL: https://arxiv.org/abs/2010.02502.