

DPM 的艺术

郑镇鑫

blair.star@163.com

目录

1 基础准备	2
1.1 随机变量的期望	2
1.2 随机变量的变换的期望	2
1.3 通过随机变量替换计算变换的期望	2
1.4 一些简单变换的期望和方差	3
1.5 随机变量的变换的概率分布	3
1.6 两个独立随机变量之和的概率分布	4
1.7 高斯分布的性质	4
1.8 log 函数的性质	5
1.9 蒙特卡洛积分 (Monte Carlo Integral)	6
1.10 概率密度函数的性质	6
1.11 贝叶斯公式 (Bayes Formula) 和全概率公式 (Total Probability Formula)	6
1.12 KL 散度及其“条件上限”	7
2 隐变量模型的复杂度及采样方法	9
2.1 更复杂的概率分布	9
2.2 并不困难的采样	10
3 三种隐变量模型及相应的参数学习方法	12
3.1 Mixture Model 及 EM 算法	12
3.2 VAE 模型及其学习方法	13
3.3 DPM 模型及其学习方法	15
3.3.1 DPM 模型的形式及其 Lower Bound	15
3.3.2 q 概率模型和 Lower Bound 的简化	18
3.3.3 三种优化 (预测) 方式	22
3.3.4 优化 MLE	24
3.4 DPM 模型的进一步分析	24
3.4.1 概括重要的结论	24
3.4.2 进一步理解目标函数	25
3.4.3 理解噪声分布向数据分布转变的过程	27
3.4.4 $q(z_{t-1} z_t)$ 概率分布的特点	27
3.4.5 $q(z_{t-1} z_t)$ 逆变换的输入敏感度	30
3.4.6 $p(z_{t-1} z_t)$ 拟合误差对逆变换的影响	32
3.4.7 压缩映射 $q(z_{t-1} z_t)$ 的不动点	33
3.4.8 DPM 模型设计要点	35
3.4.9 DPM 模型的独特之处	35
3.4.10 是否可通过“逆卷积”恢复 $q(x)$	35
3.5 融合 DPM 模型和 VAE 模型	36

1 基础准备

1.1 随机变量的期望

对于随机变量 $X \sim p_x(x)$, X_1, X_2, \dots, X_N 为相应的样本, 则期望

$$E(X) \triangleq \int x p_x(x) dx \approx \frac{1}{N} \sum_{i=1}^N X_i \quad (1)$$

上式的前半部分给出了期望的定义, 后半部分也给出了期望的一种近似的计算方式。近似误差与样本数量及 $p_x(x)$ 的复杂度相关。样本越多, 近似误差越小。 $p_x(x)$ 复杂度越低, 近似误差越小。

1.2 随机变量的变换的期望

假设存在一个随机变量 Y , 对 Y 执行一个变换 f , 得到一个新的随机变量 Z , $Z = f(Y)$, 新随机变量 Z 的期望可通过如下公式计算:

$$E(Z) = \underbrace{\int z p_z(z) dz}_{Z \text{ 的期望}} = \underbrace{\int f(y) p_y(y) dy}_{Y \text{ 的变换的期望}} \quad (2)$$

上述公式告诉我们, 要计算 Z 的期望, 可借助 Y 的概率密度分布和变换函数 $f(y)$ 计算得到, 而无需知道 Z 的概率密度分布。

假设 Y_1, Y_2, \dots, Y_N 是服从 $p_y(y)$ 的样本, 则 $f(Y_1), f(Y_2), \dots, f(Y_N)$ 是服从 $p_z(z)$ 的样本, 根据期望的近似计算方式, 则有

$$E(Z) \approx \frac{1}{N} \sum_{i=1}^N f(Y_i) \approx \frac{1}{N} \sum_{i=1}^N f(Y_i) \quad (3)$$

$$E_{p_y(y)}(f(Y)) \triangleq \int f(y) p_y(y) dy \approx \frac{1}{N} \sum_{i=1}^N f(Y_i) \quad (4)$$

式(3)告诉我们: 要计算 “ Z 的近似期望”, 无需从变换后的变量 Z 中采样样本, 只需从 Y 中采样样本, 然后把样本进行变换, 并求平均值, 即可得到 “ Z 的近似期望”。

式(4)告诉我们: 要计算 “ Y 的变换的期望的近似”, 只需从 Y 中采样样本, 然后把样本进行变换, 并求平均值, 即可得到 “ Y 的变换的期望的近似”。

注意, f 可以是任意的连续光滑的函数, 不局限于一一映射。

1.3 通过随机变量替换计算变换的期望

假设存在两个变换 f 和 g , 存在如下的变换关系

$$Z = f(Y) \quad Y = g(X) \quad Z = g(f(X))$$

$$X \sim p_x(x) \quad Y \sim p_y(y) \quad Z \sim p_z(z)$$

根据公式(2), 可得到

$$E(Z) = \int z p_z(z) dz = \int f(y) p_y(y) dy \quad (5)$$

$$E(Z) = \int z p_z(z) dz = \int f(g(x)) p_x(x) dx \quad (6)$$

于是, 进一步可得到

$$\int f(y) p_y(y) dy = \int f(g(x)) p_x(x) dx \quad (7)$$

上式左边是 “ Y 的变换的期望”, 变换为 $f(Y)$; 右边为 “ X 的变换的期望”, 变换为 “ $f(g(X))$ ”。

上式告诉我们：要计算“ Y 的变换的期望”，如果 Y 与变量 X 存在变换关系，则可通过计算“ X 的变换的期望”得到。由于与积分运算中“变量替换”较相似，所以上述计算“ Y 的变换的期望”的方式称之为“随机变量替换”。

1.4 一些简单变换的期望和方差

X, Y 是多维随机变量，

$$E(\alpha X) = \alpha E(X) \quad (8)$$

$$E(X + Y) = E(X) + E(Y) \quad (9)$$

$$D(X) = E(X^2) - [E(X)]^2 \quad (10)$$

$$D(\alpha X) = \alpha^2 D(X) \quad (11)$$

当 X, Y 相互独立时，

$$D(X + Y) = D(X) + D(Y) \quad (12)$$

1.5 随机变量的变换的概率分布

X 是一个随机变量，服从概率分布 $p_x(x)$ ，对 X 执行一个变换 f ，得到一个新的随机变量 Y ， $Y = f(X)$ ，新变量 Y 服从某个概率分布。当 f 是一一映射时（可逆的）。 Y 的概率密度函数可通过如下方式计算：

$$p_y(y) = p_x(x) \frac{1}{|f'(x)|} \quad \text{where } x = f^{-1}(y) \quad f' \text{ is derivative of } f \quad (13)$$

上式告诉我们，要计算 $Y = y$ 的概率密度 $p_y(y)$ ，可通过如下步骤计算：

- 计算 y 的逆变换得到 $x(x = f^{-1}(y))$ ；
- 计算 x 的概率密度 $p_x(x)$ ；
- 计算 x 的导数 $f'(x)$ ，然后乘以修正项 $\frac{1}{f'(x)}$ 。

为什么需要乘以修正项呢？因为当使用 f 对 x 的一个线段执行变换时，变换后对应的 y 线段会拉长或缩短，拉长或缩放的倍数为 $|f'(x)|$ ，而这两个线段的概率是相等的，所以需要除以导数。

当 X 和 X 是多维随机变量时，有类似的关系：

$$p_y(y) = p_x(x) \frac{1}{|\det(J(x))|} \quad \text{where } x = f^{-1}(y) \quad J \text{ is Jacobian matrix of } f \quad (14)$$

雅可比矩阵的行列式 (Jacobian determinant) 度量的是变换前后单位面积 (超体积) 的缩放倍数。上述结论

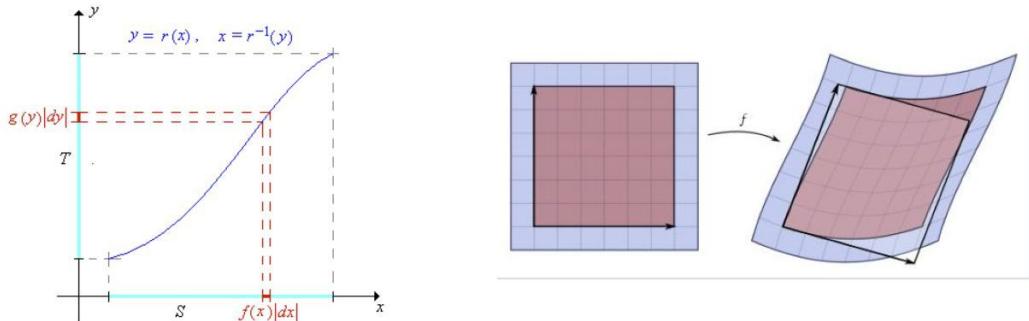


图 1: pdf of random variable transformation

的证明可看 [6]。

1.6 两个独立随机变量之和的概率分布

假设 X, Y 是两个独立随机变量, 分别服从概率分布 $p_x(x)$ 和 $p_y(y)$, 并且有相同的定义域, 则 $Z(Z = X + Y)$ 的概率分布 $p_z(z)$ 为两个概率分布的卷积, 即

$$p_z(z) = (p_x \circledast p_y)(z) \quad (15)$$

$$p_z(z) = \int p_x(\tau)p_y(z - \tau)d\tau \quad \text{when } X, Y \text{ is 1D} \quad (16)$$

举个例子, 如图 2。 X 是一维随机变量的概率分布, Y 是均值为 0 的高斯分布, 两个随机变量相加, 相当于对 X 的概率分布作高斯模糊。

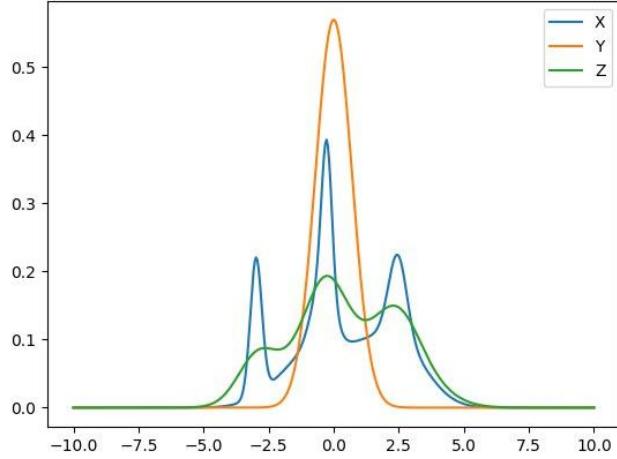


图 2: Sum of Two Independent Random Variables

关于此结论的证明, 可看 [6]。

1.7 高斯分布的性质

如果 X 是服从高斯分布的随机变量, a 和 b 是实数, 那么 $aX + b \sim \mathcal{N}(a\mu + b, a^2\Sigma^T\Sigma)$ 。

$$X \sim \mathcal{N}(\mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det(\Sigma)}} \exp \left\{ -\frac{1}{2}(x - \mu)^T \Sigma^{-1} (x - \mu) \right\} \quad (17)$$

假设 $p(x)$ 和 $q(x)$ 是随机变量 X 的两个不同的分布, $p(x) = \mathcal{N}(\mu_p, \Sigma_p)$, $q(x) = \mathcal{N}(\mu_q, \Sigma_q)$, 则 p 与 q 之间的 KL 散度有

$$KL(p||q) = \frac{1}{2} \left(\text{tr}(\Sigma_q^{-1} \Sigma_p) - k + (\mu_q - \mu_p)^T \Sigma_q^{-1} (\mu_q - \mu_p) + \log\left(\frac{\det(\Sigma_q)}{\det(\Sigma_p)}\right) \right) \quad k \text{ is dimension} \quad (18)$$

特别地, 当 $\Sigma_p = \Sigma_q = \sigma I$ 时,

$$KL(p||q) = \frac{1}{2} \frac{\|\mu_p - \mu_q\|^2}{\sigma} \quad (19)$$

假设 $f(x)$ 和 $g(x)$ 是两个高斯函数, 那么它们的积也是高斯函数。下面给出的标量的形式, 各分量相互独

立的向量形式可类推。详细证明可看 [4]。

$$\begin{aligned}
 f(x) &= \frac{1}{\sqrt{2\pi}\sigma_f} \exp\left\{-\frac{(x-\mu_f)^2}{2\sigma_f^2}\right\} & g(x) &= \frac{1}{\sqrt{2\pi}\sigma_g} \exp\left\{-\frac{(x-\mu_g)^2}{2\sigma_g^2}\right\} \\
 f(x) \times g(x) &= S \times \frac{1}{\sqrt{2\pi}\sigma_{fg}} \exp\left\{-\frac{(x-\mu_{fg})^2}{2\sigma_{fg}^2}\right\} \\
 \text{where } \mu_{fg} &= \frac{\mu_f\sigma_g^2 + \mu_g\sigma_f^2}{\sigma_f^2 + \sigma_g^2} & \sigma_{fg} &= \frac{\sigma_f^2\sigma_g^2}{\sigma_f^2 + \sigma_g^2} \\
 S &= \frac{1}{\sqrt{2\pi}(\sigma_f^2 + \sigma_g^2)} \exp\left\{-\frac{(\mu_f - \mu_g)^2}{2(\sigma_f^2 + \sigma_g^2)}\right\}
 \end{aligned} \tag{20}$$

1.8 log 函数的性质

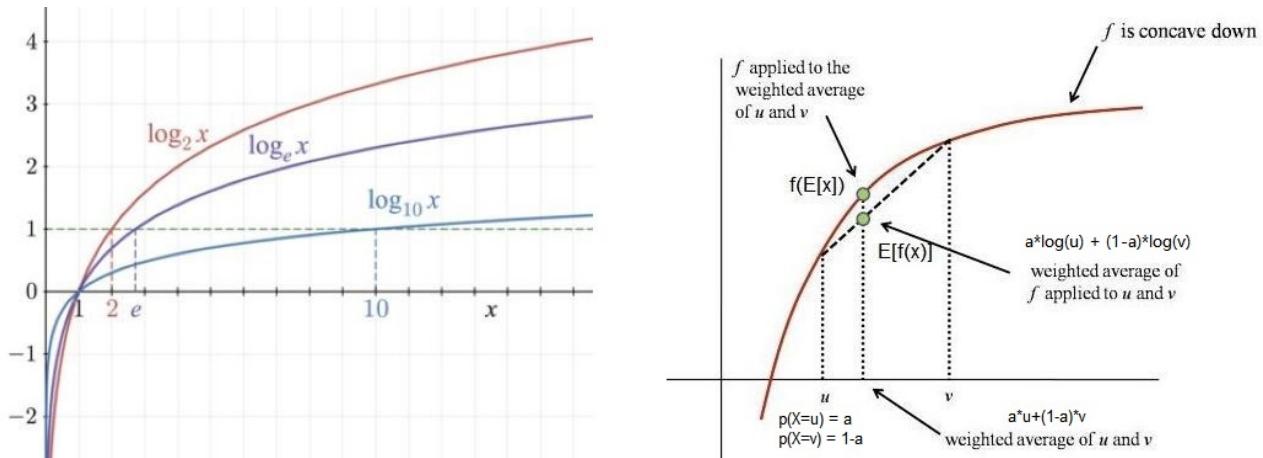


图 3: log function and concave function

$$\begin{aligned}
 \log\{f(x)g(x)\} &= \log f(x) + \log g(x) \\
 \log\left\{\frac{f(x)}{g(x)}\right\} &= \log f(x) - \log g(x) \\
 \log\{\exp\{f(x)\}\} &= f(x) \\
 \underbrace{\int p(x) \log(x) dx}_{E(\log(X))} &\leq \underbrace{\log \int x p(x) dx}_{\log(E(X))}
 \end{aligned} \tag{21}$$

- 单调递增
- 是 Concave, 可应用 Jensen 不等式。有两种理解方式。第一种方式, $\log(E(X))$ 是“曲线上的点”, $E(\log(X))$ 是“割线上的点”, 对于 Concave 函数, “曲线上的点”大于“割线上的点”。第二种方式, $\log(E(X))$ 是“期望的 log 变换”, $E(\log(X))$ 是“log 变换的期望”, “变换的期望”小于“期望的变换”。
- 消除掉指数族概率分布中的 Exp 项。指数族概率分布常用于概率模型建模。
- 简化累乘函数的求导

指数族概率分布

$$p_\theta(x) = g(\theta)h(x) \exp\{\boldsymbol{\eta}(\theta)\mathbf{u}(x)\} \tag{22}$$

1.9 蒙特卡洛积分 (Monte Carlo Integral)

准确地计算积分一般比较困难，可通过蒙特卡洛积分的方法进行近似计算。

对于任意的积分 $\int f(x)dx$, 可进行如下变换

$$\int f(x)dx = \int f(x) \frac{p_x(x)}{p_x(x)} dx = \int \frac{f(x)}{p_x(x)} p_x(x) dx \quad (23)$$

其中 $f(x)$ 为任意的函数, $p_x(x)$ 为任意的概率分布。

$\int \frac{f(x)}{p_x(x)} p_x(x) dx$ 可看做一个随机变量的变换的期望, 其中随机变量为 $X(X \sim p_x(x))$, 变换为 $\frac{f(x)}{p_x(x)}$ 。所以, 根据变换的期望的近似计算方式(4), 可得

$$\int f(x)dx = E_{p_x(x)} \left(\frac{f(X_i)}{p_x(X_i)} \right) \approx \frac{1}{N} \sum_{i=0}^N \frac{f(X_i)}{p_x(X_i)} \quad (24)$$

上述公式即蒙特卡洛积分。

近似误差的特点与期望的近似误差相同, 即跟样本数量和随机变量 $\frac{f(X)}{p_x(X)}$ 的概率分布的复杂度有关。

1.10 概率密度函数的性质

对任意的概率密度函数 $p(x)$, 其具备下面两个性质

- 对任意的 $X, p(x) \geq 0$ 。
- 定义域内积分和为 1, $\int p(x)dx = 1$ 。

反过来, 如果一个函数 $f(x)$, 要成为一个概率密度函数, 其必须满足上面两个条件。

上述的条件表示, 普通的神经网络 (比如 CNN 或 Transformer) 不能作为概率分布, 因为神经网络的输出有可能为负数, 并且积分和不为 1。

但可作一些改造, 以满足上述的条件。

- 对神经网络做简单的指数变换 $\exp(-f(x))$, 使其输出为正数, 满足条件 1。
- 求改造后函数的积分值, 然后除以积分值 (此积分值常称为 Partition Function), 使其满足条件 2。

$$p(x) = \frac{\exp(-f(x))}{Z} \quad \text{where } Z = \int \exp(-f(x))dx \quad (25)$$

通过上述方式建立的模型称之为 Energy-Based Model。在实际中, $f(x)$ 一般非常复杂, 所以 Z 的计算也非常困难。但也存在一些方法, 可在免计算 Partition Function 的条件下, 对概率分布进行学习并生成新样本。

1.11 贝叶斯公式 (Bayes Formula) 和全概率公式 (Total Probability Formula)

X, Y 是两个随机变量。 $p_x(x)$ 是 X 的概率分布; $p_y(y)$ 是 Y 的概率分布; $p_{xy}(x, y)$ 是 X, Y 的联合概率分布; $p_{x|y}(x|y)$ 是 Y 已知确定的条件下, X 的概率分布; $p_{y|x}(y|x)$ 是 X 已知确定的条件下, Y 的概率分布。贝叶斯公式给出如下的关系

$$p_{xy}(x, y) = p_{x|y}(x|y)p_y(y) \quad (26)$$

$$p_{xy}(x, y) = p_{y|x}(y|x)p_x(x) \quad (27)$$

全概率公式给出如下的关系

$$p_x(x) = \int p_{xy}(x, y)dy \quad (28)$$

$$p_y(y) = \int p_{xy}(x, y)dx \quad (29)$$

结合两个关系，可得

$$p_x(x) = \int p_{xy}(x|y)p(y)dy \quad (30)$$

$$p_y(x) = \int p_{xy}(y|x)p(x)dx \quad (31)$$

上述关系可以这么理解：要计算 X 的概率分布 $p_x(x)$ ，可借助 X 的“条件概率分布 $p_{x|y}(x|y)$ ”计算，但得考虑所有可能的条件，并以条件发生的概率 $p_y(y)$ 作为权重，综合平均所有的“条件概率分布”。

根据概率密度函数的性质，可得

$$\begin{aligned} \int p_{xy}(x,y)dxdy &= 1 \\ \int p_{x|y}(x|y)dx &= 1 \quad \text{for any } y \text{ value} \\ \int p_{y|x}(y|x)dy &= 1 \quad \text{for any } x \text{ value} \end{aligned} \quad (32)$$

上述的概率分布特意加上了下标，目的是为了从函数角度对各项进行的区分。比如， $p_x(x)$ 和 $p_y(x)$ 是两个不同的函数，如果不带下标， $p(x)$ 和 $p(y)$ 容易误认为是同一个函数，只是自变量符号不同而已。但为了让形式更简洁，在不容易混淆的情况下，不会标示出来。

另外， $p_{xy}(x,y)$ 、 $p_{x|y}(x|y)$ 及 $p_{y|x}(y|x)$ 都是关于 x, y 的函数，但却是三个不同的函数。三个函数具备不同的性质（式 32）。

1.12 KL 散度及其“条件上限”

给定两个概率分布 $p(x)$ 和 $q(x)$ ，KL 散度的定义如下：

$$KL(p\|q) = \int p(\mathbf{x}) \log \frac{p(\mathbf{x})}{q(\mathbf{x})} dx \quad (33)$$

$$KL(q\|p) = \int q(\mathbf{x}) \log \frac{q(\mathbf{x})}{p(\mathbf{x})} dx \quad (34)$$

可以看出， $KL(p\|q)$ 与 $KL(q\|p)$ 并不相等。

当两个概率分布 $p(x)$ 和 $q(x)$ 比较接近时， $KL(p\|q)$ 和 $KL(q\|p)$ 的值趋向于 0。当两个概率分布 $p(x)$ 和 $q(x)$ 差别较大时， $KL(p\|q)$ 和 $KL(q\|p)$ 的值趋向于无穷大。

假设 $p_\theta(x)$ 是参数化的概率模型， $q(x)$ 是已知确定的概率分布。以 $KL(p_\theta\|q)$ 或 $KL(q\|p_\theta)$ 为目标函数，优化 θ 能使 $p_\theta(x)$ 接近于 $q(x)$ 。虽然 $KL(p_\theta\|q)$ 与 $KL(q\|p_\theta)$ 两个目标函数有所差别，但优化的结果基本相似。

$$\min_{\theta} KL(p_\theta\|q) \iff \min_{\theta} KL(q\|p_\theta) \quad (35)$$

KL 散度存在如下的“条件上限”。

$$\begin{aligned} KL(p\|q) &= \int p(x) \log \frac{p(x)}{q(x)} dx \\ &= \int p(x) \log \frac{p(x)}{\int q(x|z)q(z)dz} dx \\ &= \int q(z)dz \int p(x) \log p(x)dx - \int p(x) \log \int q(x|z)q(z)dz dx \quad \text{where } \int q(z)dz = 1 \\ &\leq \iint q(z)p(x) \log p(x)dxdz - \int p(x) \int q(z) \log q(x|z)dz dx \quad \text{apply jensen inequality} \\ &= \iint q(z)p(x) \log p(x)dxdz - \iint q(z)p(x) \log q(x|z)dxdz \\ &= \iint q(z)p(x)(\log p(x) - \log q(x|z))dxdz \\ &= \iint q(z)p(x) \log \frac{p(x)}{q(x|z)} dxdz \\ &= \int q(z)KL(p(x)\|q(x|z))dz \triangleq KL_{up}(p(x)\|q(x|z)) \end{aligned} \quad (36)$$

$KL(p(x)\|q(x|z))$ 度量的是概率分布 $p(x)$ 与条件概率分布 $q(x|z)$ 之间的相似性。上限 $KL_{up}(p\|q)$ 则是考虑了所有的条件，并以条件发生的概率作为权重，计算其平均相似度。

在一些优化任务中， $KL(p\|q)$ 没有明确的解析式，难以进行优化。如果 $KL(p(x)\|q(x|z))$ 有明确的解析式，则可通过优化 $KL_{up}(p(x)\|q(x|z))$ ，实现对 $KL(p\|q)$ 的优化。当 $KL_{up}(p(x)\|q(x|z))$ 趋向于 0 时， $KL(p\|q)$ 也趋向于 0，两个概率分布趋向于一致。

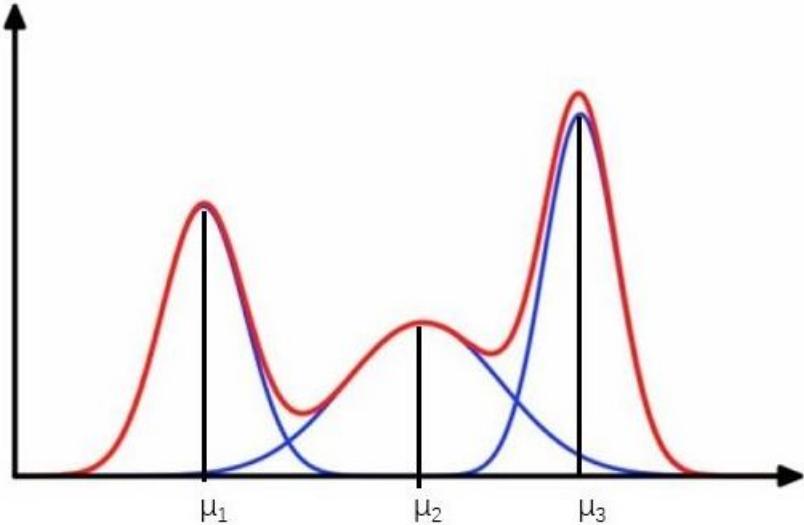


图 4: mixture gaussian model

2 隐变量模型的复杂度及采样方法

2.1 更复杂的概率分布

对于生成式概率模型，给定一批样本 $\{X_i\}_{i=1}^N$, $X_i \sim p_{data}(x)$, 希望学习一个新的概率模型 $p_{model}(x)$, 使 $p_{model}(x)$ 与 $p_{data}(x)$ 尽量接近，同时能从 $p_{model}(x)$ 快速采样得到新样本。一般情况下，希望 $p_{model}(x)$ 能有更高的自由度，能表示更复杂的函数。

对 $p_{data}(x)$ 进行建模，可分为两种方式。

1. 使用常见的概率分布直接建模，比如使用 $N(\mu, \sigma)$ 构建 $p_{model}(x)$ 。此方式的优点是参数估计简单，采样容易，但概率分布的形式较简单，难以适配复杂的问题。
2. 通过引入引变量，间接构建 $p_{model}(x)$ 。此方式分别参数化 $p(x|z)$ 和 $p(z)$ ，然后利用 Bayes 公式和全概率公式，对联合概率分布求边际分布，从而得到 $p_{model}(x)$ 。此分布有更高的复杂度，能拟合复杂的函数。

$$p_{model}(x) = \int p(x, z) dz = \int p(x|z)p(z) dz \quad (37)$$

下面举一个混合高斯模型 (Mixture Gaussian) 的例子。给定一批数据 $\{X_i\}_{i=1}^N$ ，希望通过 Mixture Gaussian 模型学习这批数据所服从的概率分布 $p(x)$ 。

混合高斯模型包含观测变量 X 和隐变量 Z , X 为连续变量 ($X \in R$), Z 为三个值的离散随机变量，假设 $p(x|z)$ 服从高斯分布， $p(z)$ 服从如下的离散分布。

$$p(z = \alpha_i) = \pi_i \quad i \in \{0, 1, 2\} \quad (38)$$

$$p(x|z = \alpha_i) \sim \mathcal{N}(\mu_i, \sigma_i^2) \quad i \in \{0, 1, 2\} \quad (39)$$

根据 bayes 公式，可得 X 和 Z 的联合概率分布 $p(x, z)$

$$p(x, z) = p(x|z)p(z)$$

根据全概率公式，可得边际分布 $p(x)$

$$p(x) = \int p(x, z) dz = \sum_{i=1}^3 p(x|z = \alpha_i)p(z = \alpha_i)$$

通过上式可知：引入隐变量 Z 后， X 的概率分布变成 3 项的加权和，其中每一项均为高斯分布，权重为 π_i ，加权和的函数形式将比任意单一高斯分布的形式更加复杂。具体可看图 4，三个蓝色曲线分别为单一的高斯概率分布 $p(x|z)$ ，红色曲线为加权和的概率分布 $p_{model}(x)$ ，红色的曲线明显比任意单一的蓝色曲线更加复杂。

上述例子的隐变量只有三个取值，如果增加取值数，或者设置为连续的随机变量，那加权和的项数将更多， $p(x)$ 的分布将更加复杂，表示能力将更强。加权和的思想与较强的模型表示能力是隐变量模型的优点之一。

当 $p(x|z)$ 设置为高斯分布时， $p(x)$ 的表示类似于 RBF 神经网络（图 5），两者都是使用高斯径向基，通过学习基的位置、形状和权重系数来拟合函数。不同的是，隐变量模型会对基和权重系数进行归一化。另外，当隐变量是连续变量时，模型可以有无限多的基数量，能表示更复杂的函数。根据一些理论研究，当基的数量足够多时，RBF 能表示任意的函数。

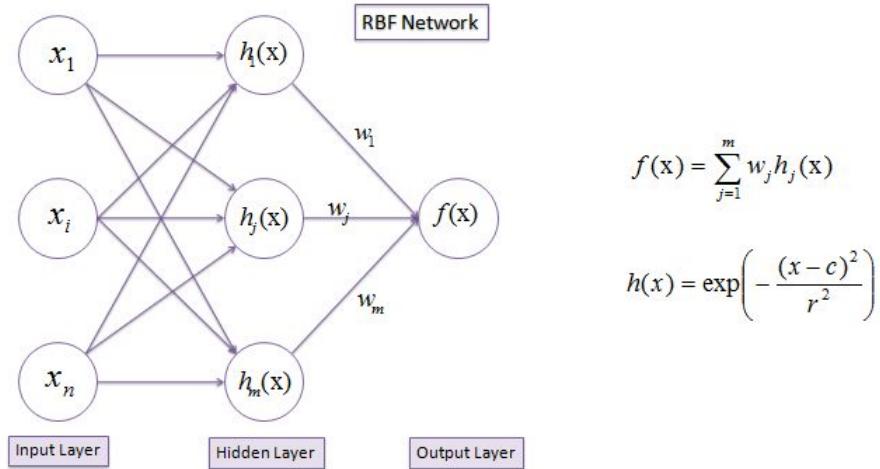


图 5: RBF Neural Network

2.2 并不困难的采样

上节提到，引入隐变量，由于累加或积分的存在， $p_{model}(x)$ 的分布变得更复杂。一般情况下，更复杂的概率分布采样将更加困难。但是通过这种方式定义的概率分布是个例外，其存在一些简单的采样方法。下面将介绍一种采样方法-祖先采样 (Ancestral Sampling)，可避免显示计算 $p_{model}(x)$ ，实现相对简单的采样。

执行祖先采样前，需利用有向图 (Directed Graph) 表示待采样的联合概率分布。借助有向图，可以更直观地执行祖先采样。有向图表示方式可参考如下的例子 (图 6)。

根据 Bayes 公式，对任意的联合概率分布 $p(x, y)$ 和 $p(x, y, z)$ 有如下关系：

$$p(x, y) = p(x|y)p(y) \quad (40)$$

$$p(x, y, z) = p(x|y, z)p(y, z) = p(x|y, z)p(y|z)p(z) \quad (41)$$

对于式(40)的联合概率分布，有两个随机变量，相应地用两个节点表示。对于 $p(x|y)$ 条件概率分布， X 依赖于 Y ，从节点 Y 向节点 X 的添加一条有向边。对于 $p(x)$ 没有依赖，无需添加边。如图 6(a) 所示。

对于式(41)的联合概率分布，有三个随机变量，相应地用三个节点表示。对于 $p(x|y, z)$ 条件概率分布， X 依赖于 Y 和 Z ，分别从 Y 和 Z 各添加一条边。对于 $p(y|x)$ ， Y 依赖于 Z ，则由 Z 向 Y 添加一条边。对于 $p(x)$ 没有依赖，无需添加边。如图 6(b) 所示。

如果在已知 Y 的条件下， X 与 Z 相互独立，则有

$$\begin{aligned} p(x|y, z) &= p(x|y) \\ p(x, y, z) &= p(x|y, z)p(y, z) = p(x|y)p(y|z)p(z) \end{aligned} \quad (42)$$

那 $p(x, y, z)$ 的表示方式如图 6(c)。与 6(b) 相比，去掉一条边。

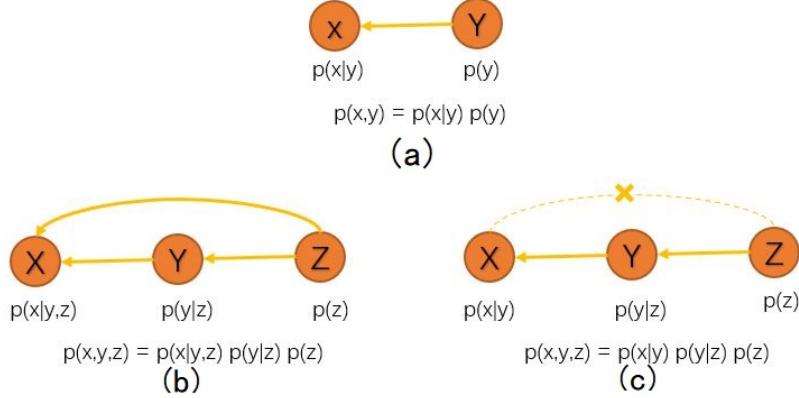


图 6: 有向图

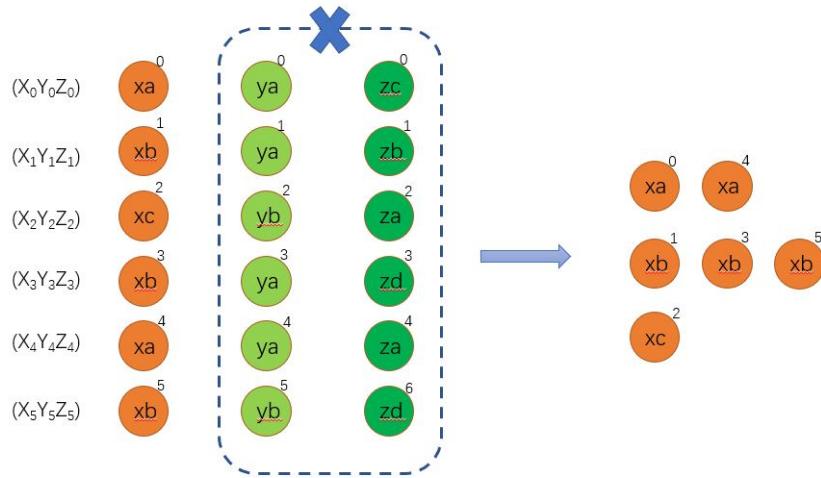


图 7: Ancestral Sampling

祖先采样 (Ancestral Sampling) 的流程如下: 以图 6(b) 举例, 从最顶层节点 Z 开始, 使用相应的概率分布 $p(z)$ 进行采样, 得到样本 Z_i 。根据边的方向往前走, 分别走到节点 Y 和 X 。由于 X 同时依赖 Y 和 Z , Y 此时尚未采样, 故暂时放下。根据 $p(y|z)$ 对 Y 节点采样, 得到样本 Y_i 。根据边的方向往前走, 走到 X , 此时依赖的 Y 和 Z 已经采样, 于是根据 $p(x|y,z)$ 对 X 进行采样, 得到 X_i 。

至此得到一个完整的样本 (X_i, Y_i, Z_i) 。此样本服从概率分布 $p(x,y,z)$ 。为什么会呢? 因为此采样流程由三步操作构成, 每步操作的概率分别为 $p(z)$ 、 $p(y|z)$ 及 $p(x|y,z)$, 这三个操作同时发生的概率为

$$p_{3op}(x = X_i, y = Y_i, Z = z_i) = p(z = X_i) p(y = Y_i | z = Z_i) p(x = X_i | y = Y_i, z = Z_i) \quad (43)$$

由 Bayes 公式可知, 上述概率即为联合分布的概率, 故 $(X_i, Y_i, Z_i) \sim p(x, y, z)$ 。重复上述的流程, 可得到一批样本 $\{(X_i, Y_i, Z_i)\}_{i=0}^N$ 。去除 Y_i 和 Z_i 分量, 可得到 $\{X_i\}_{i=0}^N$, X_i 将服从 $p(x)$ 概率分布。

为什么剩下的 $\{X_i\}_{i=0}^N$ 会服从 $p(x)$ 分布呢? 以图 7 为例子, X 变量有 3 个取值, Y 变量有 2 个取值, Z 变量 4 个取值, 采样得 6 个样本。去除 Y 和 Z 分量后, 包含 X_i 分量的多个 (X_i, Y_i, Z_i) 样本, 会得到多个重复的样本。例如, 对于 $X = xa$, 会得到第 0 和第 4 个样本, 数量为 2; 对于 $X = xb$, 会得到第 0、第 3 及第 5 个样本, 数量 3; 对于 $X = xc$, 会得到第 2 个样本, 数量 1; 此过程类似于通过全概率公式求边际分布。

至此, 我们得到一种相对简单的方法对 $p(x)$ 进行采样。关于祖先采样及概率图的更多内容可看 [7]。

上面介绍了 $p_{model}(x)$ 的表示方法, 同时也介绍了采样的方法。后面将介绍如何对 $p_{model}(x)$ 的参数进行学习估计。

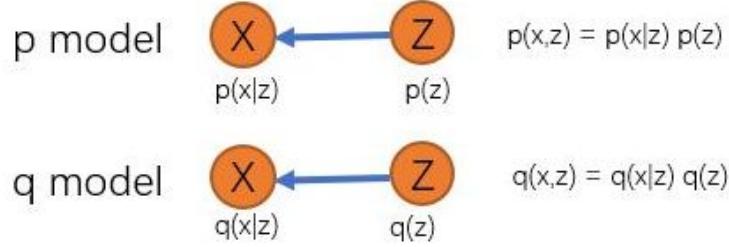


图 8: p model and q model in Mixture Model

3 三种隐变量模型及相应的参数学习方法

不同复杂度的隐变量模型会使用不同的学习方法。下面介绍三种不同复杂程度的模型及参数估计方法。

3.1 Mixture Model 及 EM 算法

对于隐变量模型, 如果 $p(z)$ 配置为离散概率分布, $p(x|z)$ 配置为指数族的概率分布, 那 $p(x)$ 可视为 Mixture Model。比如 Mixture Gaussian 模型, $p(z)$ 是离散的概率分布, 每个概率值 (π_i) 是可学习的参数, $p(x|z)$ 配置为高斯分布, 均值 (μ_i) 和方差 (σ_i) 也是可学习的参数。

对此类型的 Mixture Model, 有一个明显的特点, $p(x)$ 是一串加权和 ($p(x) = \int p(x|z)p(z)dz$), 如果优化 $\log p(x)$, \log 与 $p(x|z)$ 中的 Exp 不会直接“相乘”, 不能消除掉 Exp。如果能够消除掉 Exp, 那优化的目标函数将变得更加简单, 有利于快速优化。EM 算法能帮助解决此问题, 具体如下。

$p(x)$ 除了可使用上述的方式进行表示之外, 还可通过 Bayes 公式 ($p(x) = p(x, z)|p(z|x)$) 进行表示。Bayes 公式的形式不存在积分(累加和)的操作, 能避免上述的问题。但会另外引入一个依赖项 $p(z|x)$ 。后验概率 $p(z|x)$ 与 $p(x)$ 类似, 包含着一串加权和, 同样影响优化。EM 算法通过引入一个辅助的概率分布 $q(z)$, 然后把 $p(z|x)$ 凑成一个 KL 散度, 从而把 $p(z|x)$ 排除出待优化的目标函数(注意, 辅助概率模型 q 与 p 是两个不同的概率模型(概率空间), 但 q 和 p 有相同的定义域(样本空间))。具体方式见下述的推导。EM 算法的推导适用于离散型和连续型隐变量, 为了通用性, 下面的求和使用积分的符号表示。

为了简化, 下面的描述使用 $p(x)$ 代替 $p_{model}(x)$ 。同时, 假设 $p(x)$ 是包含 θ 的函数, 调整 θ 可得到不同的函数形式。

$p(x)$ 可进行如下的推导转化

$$\begin{aligned}
\log p(x) &= \log p(x) \int q(z) dz \\
&= \int q(z) \log p(x) dz \\
&= \int q(z) \log \frac{p(x, z)}{p(z|x)} dz \\
&= \int q(z) \log \frac{p(x, z)/q(z)}{p(z|x)/q(z)} dz \\
&= \int q(z) \log \frac{p(x, z)}{q(z)} dz - \int q(z) \log \frac{p(z|x)}{q(z)} dz
\end{aligned}
\quad \begin{aligned}
&\int q(z) dz = 1 \\
&x \text{ is independent } z \\
&\text{Bayes } p(x) = \frac{p(x, z)}{p(z|x)} \\
&\frac{q(z)}{q(z)} = 1 \\
&\log \frac{a}{b} = \log a - \log b
\end{aligned} \tag{44}$$

按如下方式定义

$$\begin{aligned}
\mathcal{L}(\theta, q) &\triangleq \int q(z) \log \frac{p(x, z)}{q(z)} dz \\
KL(q||p) &\triangleq \int q(z) \log \frac{q(z)}{p(z|x)} dz = - \int q(z) \log \frac{p(z|x)}{q(z)} dz
\end{aligned} \tag{45}$$

从而, 可将 $\log p(x)$ 表示成

$$\log p(x) = \mathcal{L}(\theta, q) + KL(q||p) \tag{46}$$

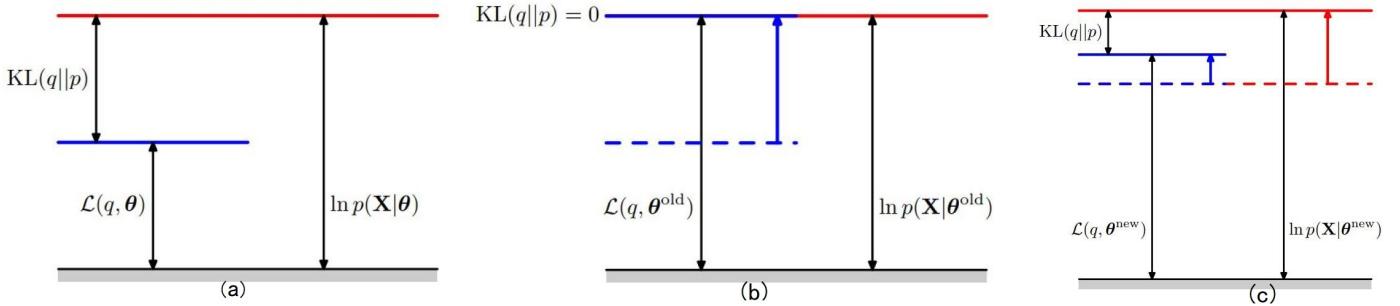


图 9: EM Lower Bound

式(46)中, q 代表任意 $q(z)$ 分布, 也就是说, 对任意的 q , 均存在上述的关系。 $KL(q||p)$ 是 KL 散度, 衡量两个概率分布的相似性, 当两个概率分布很接近时, 其值为 0; 当差别很大时, 其值趋向于正无穷。

由于 $KL(q||p)$ 总是大于等于 0, 所以 $\log p(x)$ 总是大于等于 $L(\theta, q)$, 故称 $L(\theta, q)$ 为下限 (Lower Bound)。从式(46)可以看出, 改变 q , 不会改变 $\log p(x)$, 所以 $\log p(x)$ 与 q 无关。这一点与常识相符, 毕竟这是属于两个不同的模型。但有意思的是, 改变 q 会导致 $KL(q||p)$ 和 $L(\theta, q)$ 发生变化, 并且两者的变化是“此消彼长”的关系。关于三者之间的关系, 可参考图 9(a)。

上述关于 $\log p(x)$ 的转化方式在隐变量模型中经常使用, 使用的类似的方式, 可推导出 VAE 模型和 DPM 模型的下限 (Lower Bound)。

利用式(46)的关系, EM 算法通过如下的方式对参数 θ 和 $q(z)$ 进行优化, 从而使目标函数 $\log p(x)$ 尽量大。

首先, 随机初始化参数 θ , 然后循环执行下述的两个步骤, 直至 $\log p(x)$ 没有明显的变化。

- E-Step: 固定参数 θ , 优化 $q(z)$, 使得 $KL(q||p)$ 为 0, 也就是说, 求后验概率 $p(z|x)$, 然后赋与 $q(z)$ 。优化后, $\log p(x)$ 的值不变, 但由于 $KL(q||p)$ 缩小为 0, 故 $L(\theta, q)$ 增大至 $\log p(x)$ 。如图 9(b) 所示。
- M-Step: 固定 $q(z)$, 也就是说 $q(z)$ 当作已知量, 对 $L(\theta, q)$ 中参数 θ 进行优化, 使 $L(\theta, q)$ 尽量大。优化后, 下限 $L(\theta, q)$ 增大, 同时, 由于 θ 变化, p 和 q 不再相似, 故 $KL(q||p)$ 不再为 0, 也变大了, 于是 $\log p(x)$ 增大了。如图 9(c) 所示。

上述 E-Step 有一个关键的步骤, 求解后验概率 $p(z|x)$, 如果 $p(z|x)$ 不能有效的计算, 则不能更新 $q(z)$ 。所以, 使用 EM 算法存在一个前提条件, 即 $p(z|x)$ 能高效求解。因为 $p(z|x) = \frac{p(x,z)}{p(x)}$, $p(x,z)$ 形式较为明确, 所以, 上述的条件同效于: $p(x)$ 存在明确的解析形式。对于 Mixture Model, $p(x)$ 为有限项的累加和, 故可适用。

对于 Mixture Gaussian 模型, 在优化 $L(\theta, q)$ 时, \log 与 $p(x|z)p(z)$ 直接“相乘”, 可消除掉 $p(x|z)$ 的 Exp, 剩下函数是一个二次型, 属于凹函数, 当求最大值时, 有解析解, 方便, 这样将会加快 M-step 优化的速度。

3.2 VAE 模型及其学习方法

上节, 通过 EM 算法能够估计一些类 Mixture Gaussian 模型的参数。但如 2.1 节所述, 此类的模型的表示能力有限, 不能解决复杂的问题。为了让 $p_{model}(x)$ 具备更复杂的形式, 可考虑将隐变量的先验概率分布 $p(z)$ 配置成比较复杂的分布。

VAE 模型可认为是 Mixture Gaussian 模型的升级版。对于 Mixture Gaussian 模型, 隐变量是离散的, 为了让分布更复杂, 使用连续的变量代替, 同时赋与一个概率分布 $p_z(z)$, 比如标准高斯分布。代替后, Z 的取值将变为无限多, 加权和也变成无穷多项。由于项数太多, 所以不能再逐个参数化高斯分布的均值和方差。于是, 把均值和方差设置成关于随机变量 Z 的函数 ($\tilde{\mu}_\theta(z)$ 及 $\tilde{\sigma}_\theta^2(z)$), 通过参数 θ 控制每项高斯的均值和方差。函数 $p_z(z)$ 、 $\tilde{\mu}_\theta(z)$ 和 $\tilde{\sigma}_\theta(z)$ 的自由度越高, $p(x)$ 的表示能力越强。一般情况下, $\tilde{\mu}_\theta(z)$ 及 $\tilde{\sigma}_\theta(z)$ 可设置为 CNN 或 Transformer。 $p(x)$ 的最终形式如下:

$$p(x) = \int p(x|z)p_z(z)dz \quad \text{where} \quad p(x|z) \sim \mathcal{N}(\tilde{\mu}_\theta(z), \tilde{\sigma}_\theta^2(z)), \quad p_z(z) \sim \mathcal{N}(0, I) \quad (47)$$

总结一下, VAE 模型的概率分布也是无穷多项高斯分布的加权和, 每项高斯分布的位置和形状由 $\tilde{\mu}_\theta(z)$ 和 $\tilde{\sigma}_\theta(z)$ 决定, 加权系数由标准的高斯分布决定。

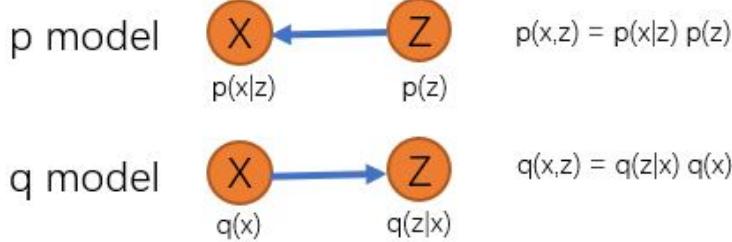


图 10: p model and q model in VAE

由于积分的存在, $p(x)$ 没有明确的解析形式, 所以不能使用 EM 算法估计参数。下面介绍另外一种参数估计方法(且称 VAE 算法), 能缓解此问题。

与 EM 算法类似, VAE 算法希望消除掉 $p(x)$ 中的积分项。通过 Bayes 公式 ($p(x) = p(x,z)|p(z|x)$) 表示能避免积分项, 但多了一个依赖项 $p(z|x)$, 后验概率 $p(z|x)$ 的计算也存在积分项, 所以也不能直接优化。VAE 算法通过引入一个辅助的概率分布 $q(z|x)$, 然后把 $p(z|x)$ 凑成一个 KL 散度, 从而把 $p(z|x)$ 排除出优化的目标函数。注意, q 和 p 是属于两个不同的概率模型(概率空间), 但 q 和 p 有相同的定义域(样本空间)。具体见下述的推导。

VAE 算法的推导与 EM 算法大体相同, 除了引入的辅助概率分布。EM 算法引入的是 $q(z)$, VAE 算法引入的是后验概率分布 $q(z|x)$ 。使用 θ 表示 $p(x)$ 包含的参数, 使用 ϕ 表示 $q(z|x)$ 包含的参数。

$p(x)$ 可进行如下的推导转化

$$\begin{aligned}
\log p(x) &= \log p(x) \int q(z|x) dz & \int q(z|x) dz = 1 \\
&= \int q(z|x) \log p(x) dz & x \text{ is independent } z \\
&= \int q(z|x) \log \frac{p(x,z)}{p(z|x)} dz & \text{Bayes } p(x) = \frac{p(x,z)}{p(z|x)} \quad (48) \\
&= \int q(z|x) \log \frac{p(x,z)/q(z|x)}{p(z|x)/q(z|x)} dz & \frac{q(z|x)}{q(z|x)} = 1 \\
&= \int q(z|x) \log \frac{p(x,z)}{q(z|x)} dz - \int q(z|x) \log \frac{p(z|x)}{q(z|x)} dz & \log \frac{a}{b} = \log a - \log b
\end{aligned}$$

按如下方式定义

$$\begin{aligned}
\mathcal{L}(\theta, \phi) &\triangleq \int q(z|x) \log \frac{p(x,z)}{q(z|x)} dz = E_{q(z|x)} \left[\log \frac{p(x,z)}{q(z|x)} \right] \\
KL(q||p) &\triangleq \int q(z|x) \log \frac{q(z|x)}{p(z|x)} dz = - \int q(z|x) \log \frac{p(z|x)}{q(z|x)} dz
\end{aligned} \quad (49)$$

从而, 可将 $\log p(x)$ 表示成

$$\log p(x) = \mathcal{L}(\theta, \phi) + KL(q||p) \quad (50)$$

由于推导过程中没有对 $q(z|x)$ 作任何的假设, 所以, 对任意的 $q(z|x)$, 均存在上述关系。

由于 $KL(q||p) \geq 0$, 所以有

$$\log p(x) \geq \mathcal{L}(\theta, \phi) \quad (51)$$

由于上述关系的存在, 所以, 可通过优化 $\mathcal{L}(\theta, \phi)$, 从而实现对 $\log p(x)$ 的优化。

为了突出 $p(x|z)$ 是包含 θ 参数的函数, $q(z|x)$ 是包含 ϕ 参数的函数, 下面的公式会在下标写出包含的参数。另外, 为了区分 $p(x)$ 和 $p(z)$ 两个函数, 使用 $p_z(x)$ 符号代替 $p(z)$ 。根据 Bayes 公式及 KL 散度的定义, 可对 $\mathcal{L}(\theta, \phi)$ 进一步转化

$$\mathcal{L}(\theta, \phi) = E_{q_\phi(z|x)} [\log p_\theta(x|z)] - KL(q_\phi(z|x)||p_z(z)) \quad (52)$$

可以看出, 上式第一项为 **q 概率模型中随机变量的变换的期望**。其中, 概率分布为 $q_\phi(z|x)$, 变换函数为 $p_\theta(x|z)$ 。

一般情况下，可以通过 Monte Carlo 采样的方式近似计算上述的期望。但由于待采样的概率分布 $q_\phi(z|x)$ 存在待优化的参数，所以不能直接近似，否则，在优化过程中，会对参数的梯度造成较大的误差，影响稳定性。

为了解决此问题，可对 q 概率模型中 $q_\phi(z|x)$ 作进一步的假设，假设 q 概率模型中的随机变量 Z 可由如下的函数变换得到。

$$z = g_\phi(x, \epsilon) = \bar{\mu}_\phi(x) + \bar{\sigma}_\phi(x)\epsilon \quad \text{where } \epsilon \sim \mathcal{N}(0, I) \quad (53)$$

其中， $\bar{\mu}_\phi(x)$ 和 $\bar{\sigma}_\phi(x)$ 为两个任意的函数，实际中，可使用 CNN 和 Transformer 替代。

可以看出，函数 $g_\phi(x, \epsilon)$ 依赖于 x ，同时也是关于随机变量 ϵ 的函数。根据 1.2 节的结论，随机变量的变换也是一个随机变量，并服从某个概率分布。因此，根据 1.7 节中的性质，可知：在 X 已知（固定）的条件下， Z 的条件概率分布 $q_\phi(z|x)$ 是一个高斯分布函数，均值为 $\bar{\mu}_\phi(x)$ ，标准差 $\bar{\sigma}_\phi(x)$ 。

根据 1.3 的结论，当计算期望时，如果随机变量可由其它随机变量变换得到，则可对期望中的随机变量进行替换。对于 $\mathcal{L}(\theta, \phi)$ 中的第一项-期望项，随机变量为 Z ，概率分布为 $q_\phi(z|x)$ ，变换为 $\log p_\theta(x|z)$ ， Z 可由 ϵ 通过 $g_\phi(x, \epsilon)$ 变换得到。于是，可对 Z 使用 ϵ 进行替换。替换后的 $\mathcal{L}(\theta, \phi)$ 变成

$$\begin{aligned} \mathcal{L}(\theta, \phi) &= E_{q(\epsilon)} [\log p_\theta(x|g_\phi(x, \epsilon))] - KL(q_\phi(z|x)||p_z(z)) \\ &\quad \text{where } q(\epsilon) = \mathcal{N}(0, I) \end{aligned} \quad (54)$$

经过变量替换后，待采样的随机变量的概率分布不再包含参数，所有的参数均在期望中的变换函数 ($\log p_\theta(x|g_\phi(x, \epsilon))$) 内部。所以，可对此项使用 Monte Carlo 积分（1.9 节）进行近似，近似后的 $\mathcal{L}(\theta, \phi)$ 如下

$$\begin{aligned} \mathcal{L}(\theta, \phi) &= \sum_{i=1}^K \log p_\theta(x|g_\phi(x, \epsilon_i)) - KL(q_\phi(z|x)||p_z(z)) \\ &\quad \text{where } \epsilon_i \sim \mathcal{N}(0, I) \quad i = 1, 2, \dots, K \end{aligned} \quad (55)$$

对于 $\mathcal{L}(\theta, \phi)$ 中的第二项，是关于 $q_\phi(z|x)$ 和 $p_z(z)$ 之间的 KL 散度，也称为一致项（Consistent Term）。由式(47)可知， $p_z(z)$ 为高斯分布函数，而 $q_\phi(z|x)$ 也为高斯分布，所以，根据 1.7 节的结论，此项具有明确的解析式，无需使用 Monte Carlo 积分近似。另外，由于 KL 散度恒大于等于 0，所以，此项在优化中的作用是，使 $q_\phi(z|x)$ 概率分布与 $p_z(z)$ 概率分布尽量相似。

在优化过程中，此项一般比较难优化，会造成较大的损失。原因如下： $p_z(z)$ 为固定的高斯分布，而 $q_\phi(z|x)$ 是一个变化的高斯分布，其均值随 X 的变化而变化，所以此项很难完全相似。后面的 DDPM 模型也会看到类似的损失项，但 DDPM 模型能较好地解决此问题。

对于第一项，在实践中，经常作如下简化：Monte Carlo 积分近似只使用一个 ϵ 样本，即使用 ϵ 的均值，同时，也把 $p(x|z)$ 的方差设置成固定。所以，可将 $\mathcal{L}(\theta, \phi)$ 中的期望项简化成如下的形式

$$E_{q(\epsilon)} [\log p_\theta(x|g_\phi(x, \epsilon))] \approx \alpha * \|x - \tilde{\mu}_\theta(\bar{\mu}_\phi(x))\|^2 \quad (56)$$

其中 α 独立于 ϕ 和 θ 的常数。于是，此项也常称之为重建项（Reconstruction Term）。

至此， $\mathcal{L}(\theta, \phi)$ 中的两项均化简成可直接优化的目标函数。

为什么要引入 $q(z|x)$ 而不是 $q(z)$ 呢？首先，对于 Lower Bound 的推导，使用 $q(z)$ 和 $q(z|x)$ 均成立。引入 $q(z|x)$ 能建立起 $x \rightarrow z$ 的关系，让 $E_{q_\phi(z|x)} [\log p_\theta(x|z)]$ 项演变成欧氏距离损失项，从而使 VAE 可作为一种降维的工具。至于是否能让 $p(x)$ 学习的更好、更接近真实的数据分布，本人尚未分析出结论，但网络上有个别回答提到，使用 $q(z|x)$ 能优化出更好的效果。

3.3 DPM 模型及其学习方法

3.3.1 DPM 模型的形式及其 Lower Bound

DPM 模型的形式

DPM 模型可视为 VAE 模型的升级版。

由上面的介绍可知以下两个结论。

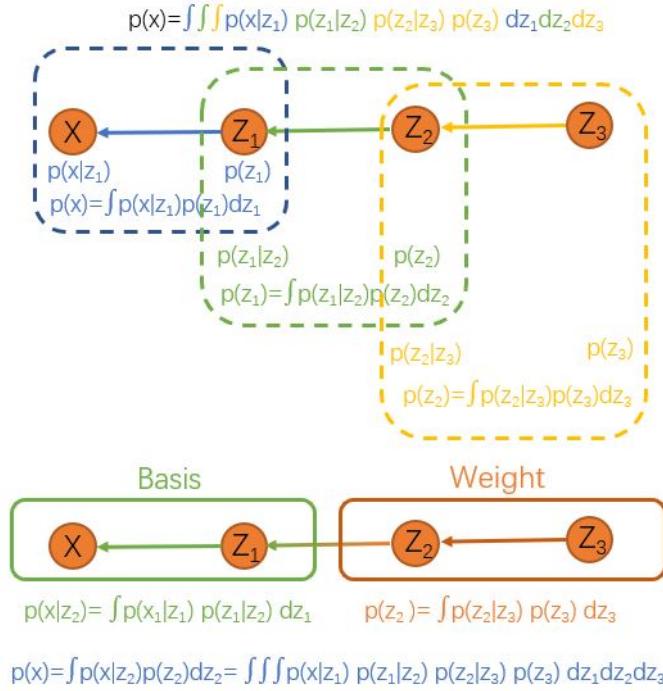


图 11: p model in DPM

- 引入隐变量，可使 $p(x)$ 的复杂度变得更高 ($p(x) = \int p(x|z)p(z)dz$)。
- 隐变量的先验概率 $p(z)$ 越复杂， $p(x)$ 的复杂度将越高。

在 VAE 中， $p(z)$ 配置成固定的高斯分布，较为简单。为了进一步提升 $p(x)$ 的复杂度，可考虑将 $p(z)$ 配置成更复杂的分布。根据结论 (1)，引入隐变量可提升复杂度，于是可引入第二个隐变量。引入后的 $p(x)$ 形式为

$$\begin{aligned} p(x) &= \int p(x|z_1)p(z_1) dz_1 \quad p(z_1) = \int p(z_1|z_2)p(z_2) dz_2 \\ \Rightarrow p(x) &= \underbrace{\int p(x|z_1)}_{basis} \underbrace{\int p(z_1|z_2)p(z_2) dz_1 dz_2}_{weight} = \iint p(x|z_1)p(z_1|z_2)p(z_2) dz_1 dz_2 \end{aligned} \quad (57)$$

可以看出，引入后，**加权和的系数**变为 $\int p(z_1|z_2)p(z_2)dz_1$ 。系数的复杂度变高，表示的函数的复杂度也随之提升。类推可引入更多的隐变量，从而，进一步提升 $p(x)$ 的复杂度。

对于三个隐变量以上的模型形式，可从另一个角度进行解释。对于三个隐变量的联合概率分布，可通过如下的形式表示。

$$\begin{aligned} p(x) &= \int p(x|z_1) \left\{ \int p(z_1|z_2)p(z_2|z_3)p(z_3) dz_2 dz_3 \right\} dz_1 \\ &= \iiint p(x|z_1)p(z_1|z_2)p(z_2|z_3)p(z_3) dz_1 dz_2 dz_3 \\ &= \underbrace{\int \left\{ \int p(x|z_1)p(z_1|z_2) dz_1 \right\}}_{basis} \underbrace{\left\{ \int p(z_2|z_3)p(z_3) dz_3 \right\}}_{weight} dz_2 \end{aligned} \quad (58)$$

可以看出，第一项 basis 是关于 z_2 的函数， z_2 的每个取值对应一个基函数。每个基函数是关于 z_1 隐变量的加权和，所以比单一高斯径向基更复杂。第二项则是基函数的权重系数。可通过图 11 可更好地理解。

下面以 T 个隐变量的模型进行介绍。在 DDPM 论文中，隐变量个数 $T = 1000$ 。

T 个隐变量的 $p(x)$ 形式如下

$$p(x) = \int p(x|z_1) \prod_{t=2}^T p(z_{t-1}|z_t)p(z_T) dz_{1:T} = \int p(x|z_1)p(z_1|z_2) \cdots p(z_{T-1}|z_T)p(z_T) dz_{1:T} \quad (59)$$

与 VAE 相似，也把 $p(x|z_1)$ 和 $\{p(z_{t-1}|z_t)\}_{t=2}^T$ 均配置为高斯分布，其形状和位置通过学习确定。 $p(z_T)$ 设置标准高斯分布。

$$\begin{aligned} p(x|z_1) &\sim \mathcal{N}(\mu_{\theta_1}(z_1), \sigma_{\theta_1}(z_1)) \\ p(z_{t-1}|z_t) &\sim \mathcal{N}(\mu_{\theta_t}(z_t), \sigma_{\theta_t}(z_t)) \quad t \in \{2, \dots, T\} \\ p(z_T) &\sim \mathcal{N}(0, I) \end{aligned} \quad (60)$$

与 VAE 不同的是，隐变量的维度 z_t 与 x 维度是相同的。实际使用中，经常也把方差固定，即 $\sigma_{\theta_t}(z_t) = \sigma_t^2$ 。

$$p(x) = \iiint p(x, z_{1:T}) dz_{1:T} \quad (61)$$

把式(59)与全概率公式 (式(61)) 相比较，可得

$$p(x, z_{1:T}) = p(x|z_1) \prod_{t=2}^T p(z_{t-1}|z_t) p(z_T) = p(x|z_1)p(z_1|z_2) \cdots p(z_{T-1}|z_T)p(z_T) \quad (62)$$

同时，根据 Bayse 公式有

$$p(x, z_{1:T}) = p(x|z_{1:T}) \prod_{t=2}^T p(z_{t-1}|z_{t:T}) p(z_T) = p(x|z_{1:T})p(z_1|z_{2:T}) \cdots p(z_{T-1}|z_T)p(z_T) \quad (63)$$

对比上述两式，可得

$$\begin{aligned} p(x|z_{1:T}) &= p(x|z_1) \\ p(z_{t-1}|z_{t:T}) &= p(z_{t-1}|z_t) \quad t \in \{2, \dots, T\} \end{aligned} \quad (64)$$

由上述的条件独立性可知， x, z_1, \dots, z_T 构成一条 Markov Chain($x \leftarrow z_1 \cdots z_{T-1} \leftarrow z_T$)。通过图 11更直观地理解此过程。

推导出 $\log p(x)$ 的 Lower Bound

对于 $\log p(x)$ 的优化，与 VAE 的思路相似。

首先，要消除积分项，所以应用 Bayes 公式进行替换。

$$p(x) = \frac{p(x, z_{1:T})}{p(z_{1:T}|x)} \quad (65)$$

其次，要消除掉 p 模型的后验概率项。因此，引入一个辅助概率模型 q ，并选用一个关于 $z_{1:T}$ 的概率分布，可以是 $q(z_{1:T})$ 或者 $q(z_{1:T}|x)$ ，目的是为了凑成 KL 散度项。参考 VAE 的结论，选用 $q(z_{1:T}|x)$ ，让 $z_{1:T}$ 依赖于 x ，能优化得到更好的结果。

与 VAE 的推导类似，可把 $\log p(x)$ 转化成 Lower Bound 和 KL 散度之和。

$$\begin{aligned} \log p(x) &= \log p(x) \int q(z_{1:T}|x) dz_{1:T} & \int q(z_{1:T}|x) dz_{1:T} &= 1 \\ &= \int q(z_{1:T}|x) \log p(x) dz_{1:T} & x \text{ is independent of } z \\ &= \int q(z_{1:T}|x) \log \frac{p(x, z_{1:T})}{p(z_{1:T}|x)} dz_{1:T} & \text{Bayes} \\ &= \int q(z_{1:T}|x) \log \frac{\frac{p(x, z_{1:T})}{q(z_{1:T}|x)}}{\frac{p(z_{1:T}|x)}{q(z_{1:T}|x)}} dz_{1:T} & \frac{q(z_{1:T}|x)}{q(z_{1:T}|x)} = 1 \\ &= \int q(z_{1:T}|x) \log \frac{p(x, z_{1:T})}{q(z_{1:T}|x)} dz_{1:T} - \int q(z_{1:T}|x) \log \frac{p(z_{1:T}|x)}{q(z_{1:T}|x)} dz_{1:T} & \log \frac{a}{b} = \log a - \log b \end{aligned} \quad (66)$$

按如下方式定义

$$\begin{aligned} \mathcal{L}(\theta, \phi) &\triangleq \int q(z_{1:T}|x) \log \frac{p(x, z_{1:T})}{q(z_{1:T}|x)} dz = E_{q(z_{1:T}|x)} \left[\log \frac{p(x, z_{1:T})}{q(z_{1:T}|x)} \right] \quad \phi \text{ is parameter of } q \\ KL(q||p) &\triangleq \int q(z) \log \frac{q(z_{1:T}|x)}{p(z_{1:T}|x)} dz = - \int q(z_{1:T}|x) \log \frac{p(z_{1:T}|x)}{q(z_{1:T}|x)} dz \end{aligned} \quad (67)$$

从而，可将 $\log p(x)$ 表示成

$$\log p(x) = \mathcal{L}(\theta, \phi) + KL(q||p) \quad (68)$$

由于 $KL(q||p) \geq 0$ ，所以有

$$\log p(x) \geq \mathcal{L}(\theta, \phi) \quad (69)$$

基于上述的关系，可通过优化 $\mathcal{L}(\theta, \phi)$ ，从而实现对 $\log p(x)$ 的优化。

根据式(62)，可将 $\mathcal{L}(\theta, \phi)$ 写成

$$\mathcal{L}(\theta, \phi) = \int q(z_{1:T}|x) \log \frac{p(x|z_1) \prod_{t=2}^T p(z_{t-1}|z_t) p(z_T)}{q(z_{1:T}|x)} dz_{1:T} \quad (70)$$

上述式子是多重积分，虽然可以通过 Monte Carlo 积分近似计算，但为了减少近似误差，得办法消除一些积分。怎么消除呢？有两种思路：

- 利用全概率分布，对联合概率分布求边际分布 ($\int p(x, y) dx = p(y)$)。
- 凑 KL 散度项，两个高斯分布的 KL 散度具有明确的解析式 (1.7节)。

注意，上述 $\mathcal{L}(\theta, \phi)$ 的推导对任意 $q(z_{1:T}|x)$ 均成立，并且 $p(x)$ 的值不随 $q(z_{1:T}|x)$ 的变化而变化。所以，可对 $q(z_{1:T}|x)$ 作任何的假设，只要方便优化 $\mathcal{L}(\theta, \phi)$ 便可。

于是对 $q(z_{1:T}|x)$ 概率分布作一些假设。下面介绍 DDPM 模型中的假设方式。

3.3.2 q 概率模型和 Lower Bound 的简化

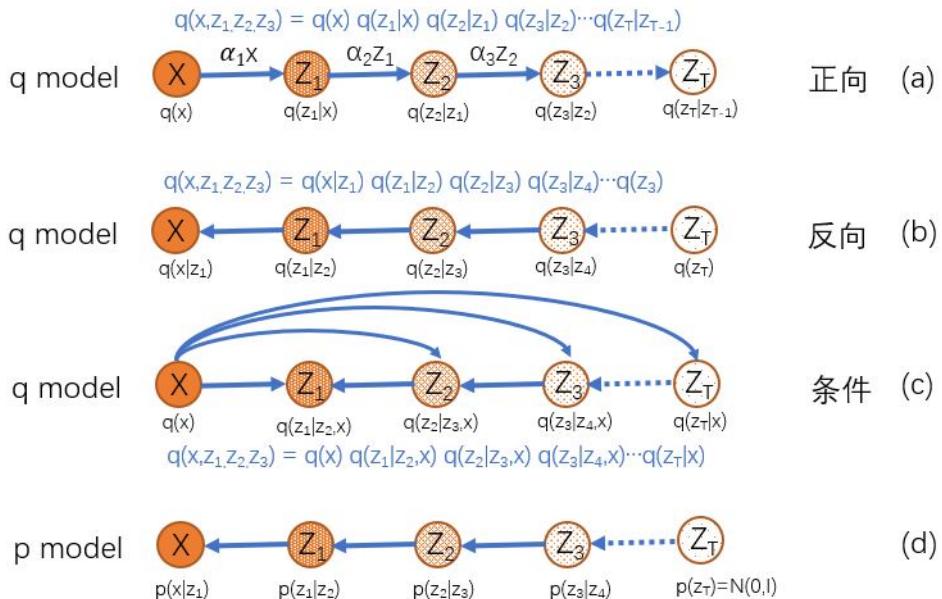


图 12: q model in DPM

假设 q 概率模型的随机变量 $X, Z_{1:T}$ 满足如下的关系

$$\begin{aligned} Z_1 &= \sqrt{\alpha_1}X + \sqrt{1-\alpha_1}\epsilon_1 \\ Z_t &= \sqrt{\alpha_t}Z_{t-1} + \sqrt{1-\alpha_t}\epsilon_t \quad t \in \{2, \dots, T\} \\ \epsilon_t &\sim \mathcal{N}(0, I) \quad t \in \{1, \dots, T\} \end{aligned} \quad (71)$$

其中， α_t 是常数，并且小于 1。

由于 $q(z_{1:T}|x)$ 是一个固定的分布，没有参数，所以， $\mathcal{L}(\theta, \phi)$ 简化成 $\mathcal{L}(\theta)$ 。

另外，假设 q 模型中的随机变量 X 的概率分布 $q(x)$ 为 $p_{data}(x)$ 。注意，概率分布 $p_{data}(x)$ 是未知的，但已知服从此分布的一批样本，这批样本就是给定的训练数据。

$$q(x) = p_{data}(x) \quad (72)$$

在 DDPM 论文中， q 的隐变量个数 $T=1000$ ， α_1 设为 0.9999， α_{1000} 设为 0.98，中间的值 α_t 通过线性插值确定。

通过分析变换的过程，可直观理解 X, Z_1, \dots, Z_T 的概率密度的变化规律。上述变换主要由两个子变换组成。

- 一个简单的可逆线性变换 $Y = \sqrt{\alpha_t} X$ 。通过 1.5 节结论可知， Y 的概率分布为

$$p_y(y) = \frac{1}{\sqrt{\alpha_t}} p_x(\frac{y}{\sqrt{\alpha_t}}) \quad (73)$$

可以看出，变换后的分布 $p_y(y)$ 可由 $p_x(x)$ 简单变换而来：把 $p_x(x)$ 函数形式收缩 $\sqrt{\alpha_t}$ 倍，幅度拉升 $\sqrt{\alpha_t}$ 倍。

- 两个独立随机变量相加。 ϵ_t 与 X 和 Z_t 均是相互独立，根据 1.6 节的结论，相加后的变量的概率分布相当于两个概率密度函数的卷积。由于 ϵ_t 是标准高斯分布，故新概率分布可认为是高斯模糊的结果。

所以，上述变换的过程可认为是概率密度“收缩拉升-模糊”的过程。图 13 是一个一维随机变量的例子， $q(x)$ 是随机生成的概率密度，经过一个变换后 ($\alpha = 0.9$)， $q(z_1)$ 的概率密度沿中心收缩拉升，同时细节变少，轮廓变粗。重复 11 个变换， $q(z_{11})$ 变为近似的标准高斯分布。

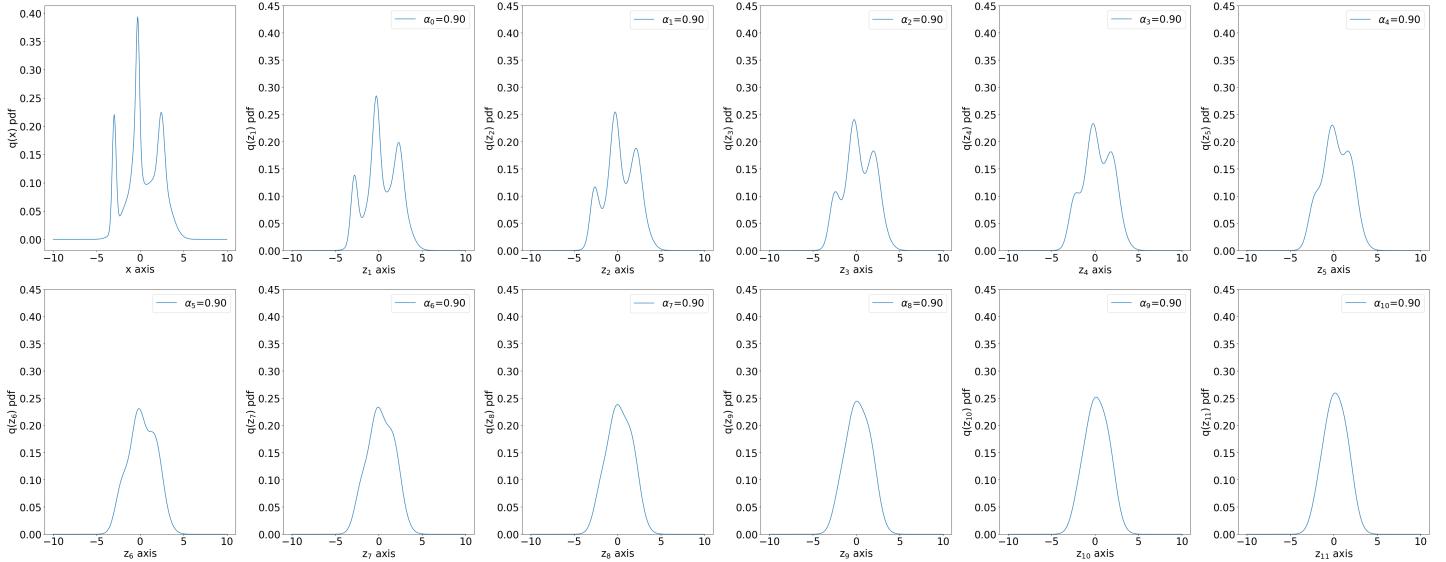


图 13: $q(z_t)$ with different t

性质 1

根据上述假设的关系，可知 $X, Z_{1:T}$ 构成一条 Markov Chain($X \rightarrow Z_1 \cdots Z_2 \rightarrow Z_T$)，并且有

$$\begin{aligned} q(z_1|x) &= \mathcal{N}(\sqrt{\alpha_1}x, 1 - \alpha_1) \\ q(z_t|z_{1:t-1}, x) &= q(z_t|z_{t-1}) = \mathcal{N}(\sqrt{\alpha_t}z_{t-1}, 1 - \alpha_t) \quad t \in \{2, \dots, T\} \end{aligned} \quad (74)$$

性质 2

利用上述的条件独立性 (式(74)) 及 Bayes 公式，可将 $q(z_{1:T}|x)$ 和 $q(x, z_{1:T})$ 展开成

$$\begin{aligned} q(z_{1:T}|x) &= \prod_{t=2}^T q(z_t|z_{t-1}, x) q(z_1|x) = \prod_{t=2}^T q(z_t|z_{t-1}) q(z_1|x) \\ q(x, z_{1:T}) &= \prod_{t=2}^T q(z_t|z_{t-1}) q(z_1|x) q(x) = q(z_T|z_{T-1}) \cdots q(z_2|z_1) q(z_1|x) q(x) \end{aligned} \quad (75)$$

在本文中，将此形式称为“正向展开”。相应的概率图结构可见图 12(a)。从性质 1 可看出，“正向展开”的条件概率分布的具备明确已知的形式。

性质 3

另外，根据 Bayes 公式，后验概率 $q(z_t|z_{t-1})$ 存在如下的关系

$$q(z_t|z_{t-1}) = q(z_{t-1}|z_t) \frac{q(z_t)}{q(z_{t-1})} \quad t \in \{2, \dots, T\} \quad (76)$$

于是，将上述的关系代入式(75)，可得 $q(z_1, z_2, z_3|x)$ 和 $q(x, z_1, z_2, z_3)$ 的第二种形式。

$$\begin{aligned} q(x, z_{1:T}) &= q(z_T|z_{T-1}) \cdots q(z_2|z_1)q(z_1|x)q(x) \\ &= \cdots q(z_2|z_3) \frac{q(z_3)}{\cancel{q(z_2)}} q(z_1|z_2) \cancel{\frac{q(z_2)}{q(z_1)}} q(x|z_1) \cancel{\frac{q(z_1)}{q(x)}} q(x) \\ &= q(z_T)q(z_{T-1}|z_T) \cdots q(z_2|z_3)q(z_1|z_2)q(z_1|x) \\ &= q(z_1|x)q(z_1|z_2)q(z_2|z_3) \cdots q(z_{T-1}|z_T)q(z_T) \end{aligned} \quad (77)$$

在本文中，将此形式称为“反向展开”。相应的概率图结构可见图 12(b)。“反向展开”的各个条件概率分布均没有明确已知的形式。

性质 4

类似地，根据 Markov Chain 的性质及 Bayes 公式，后验概率 $\{q(z_t|z_{t-1})\}_{t=2}^T$ 也存在如下的关系

$$\begin{aligned} q(z_t|z_{t-1}) &= q(z_t|z_{t-1}, x) && \text{Markov Chain} \\ &= \frac{q(z_t, z_{t-1}|x)}{q(z_{t-1}|x)} && \text{Bayes} \\ &= q(z_{t-1}|z_t, x) \frac{q(z_t|x)}{q(z_{t-1}|x)} && \text{Bayes} \end{aligned} \quad (78)$$

于是，将上述的关系代入式(75)，可得 $q(z_1, z_2, z_3|x)$ 和 $q(x, z_1, z_2, z_3)$ 的第三种形式。

$$\begin{aligned} q(z_{1:T}|x) &= q(z_T|z_{T-1}) \cdots q(z_3|z_2)q(z_2|z_1)q(z_1|x) \\ &= \cdots q(z_2|z_3, x) \frac{q(z_3|x)}{\cancel{q(z_2|x)}} q(z_1|z_2, x) \cancel{\frac{q(z_2|x)}{q(z_1|x)}} q(z_1|x) \\ &= q(z_T|x)q(z_{T-1}|z_T, x) \cdots q(z_2|z_3, x)q(z_1|z_2, x) \\ &= q(z_1|z_2, x)q(z_2|z_3, x) \cdots q(z_{T-1}|z_T, x)q(z_T|x) \\ q(x, z_1, z_2, z_3) &= q(z_1|z_2, x)q(z_2|z_3, x) \cdots q(z_{T-1}|z_T, x)q(z_T|x)q(x) \end{aligned} \quad (79)$$

在本文中，将此形式称为“条件展开”。相应的概率图结构可见图 12(c)。将在性质 6 看到，“条件展开”的条件概率分布具备明确已知的形式。

性质 5

根据随机变量 $X, Z_{1:T}$ 之间假设的关系，可推导出 $z_{2:T}$ 与 X 存在如下的关系

$$\begin{aligned} z_2 &= \sqrt{\alpha_2}(\sqrt{\alpha_1}x + \sqrt{1-\alpha_1}\epsilon_1) + \sqrt{1-\alpha_2}\epsilon_2 \\ &= \sqrt{\alpha_2\alpha_1}x + \sqrt{\alpha_2 - \alpha_2\alpha_1}\epsilon_1 + \sqrt{1-\alpha_2}\epsilon_2 \end{aligned} \quad (80)$$

在 x 已知(固定)条件下， $\sqrt{\alpha_1\alpha_2}x$ 是常数， ϵ_1 和 ϵ_2 是两个独立的标准高斯分布。所以，根据 1.7 节中高斯分布的性质，有

$$q(z_2|x) = \mathcal{N}(\sqrt{\alpha_1\alpha_2}x, 1 - \alpha_1\alpha_2) \quad (81)$$

同理，可递推得出

$$q(z_t|x) = \mathcal{N}(\sqrt{\alpha_1\alpha_2\cdots\alpha_t}x, 1 - \alpha_1\alpha_2\cdots\alpha_t) = \mathcal{N}(\sqrt{\bar{\alpha}_t}x, 1 - \bar{\alpha}_t) \quad \text{where } \bar{\alpha}_t \triangleq \prod_{j=1}^t \alpha_j \quad (82)$$

也就是说， $q(z_1|x), \{q(z_t|x)\}_{t=2}^T$ 均是高斯分布，并且这些高斯分布具有一个特点：由于 α_t 小于 1，当连乘个数增多时， $\bar{\alpha}_t$ 的值逐渐减小，于是， $q(z_t|x)$ 的均值逐渐减小，方差逐渐增大；当连乘个数足够多时（ t 足够大时）， $q(z_t|x)$ 将近似于 $\mathcal{N}(0, I)$ 。

性质 6

根据 Bayes 公式及两个高斯函数乘积的性质（1.7节），可确定 $\{q(z_{t-1}|z_t, x)\}_{t=2}^T$ 概率分布的解析式。具体如下：

$$q(z_{t-1}|z_t, x) = \frac{q(z_t|z_{t-1})q(z_{t-1}|x)}{q(z_t|x)} \quad (83)$$

由于 $q(z_t|x)$ 与 z_{t-1} 无关，所以 $q(z_t|x)$ 是一项常数，记为 K。 $q(z_t|z_{t-1})$ 是关于 z_t 的高斯概率分布，也可看作是关于 z_{t-1} 的高斯函数。 $q(z_{t-1}|x)$ 是关于 z_{t-1} 高斯概率分布。应用 1.7节的结论，两个高斯函数的乘积也是高斯函数，所以 $q(z_{t-1}|z_t, x)$ 也是高斯概率分布。推导如下：

$$\begin{aligned} q(z_{t-1}|z_t, x) &= \frac{q(z_t|z_{t-1})q(z_{t-1}|x)}{q(z_t|x)} \\ &= \frac{1}{K} \times \frac{1}{\sqrt{2\pi(1-\alpha_t)}} \exp\left\{\frac{-(z_t - \sqrt{\alpha_t}z_{t-1})^2}{2(1-\alpha_t)}\right\} \times \frac{1}{\sqrt{2\pi(1-\bar{\alpha}_{t-1})}} \exp\left\{\frac{-(z_{t-1} - \sqrt{\bar{\alpha}_{t-1}}x)^2}{2(1-\bar{\alpha}_{t-1})}\right\} \\ &= \frac{1}{K\sqrt{\alpha_{t-1}}} \times \frac{1}{\sqrt{2\pi\sigma_f^2}} \exp\left\{\frac{-(z_{t-1} - \mu_f)^2}{2\sigma_f^2}\right\} \times \frac{1}{\sqrt{2\pi\sigma_g^2}} \exp\left\{\frac{-(z_{t-1} - \mu_g)^2}{2\sigma_g^2}\right\} \\ &= \frac{S}{K\sqrt{\alpha_{t-1}}} \times \frac{1}{\sqrt{2\pi\sigma_{fg}^2}} \exp\left\{\frac{-(z_{t-1} - \mu_{fg})^2}{2\sigma_{fg}^2}\right\} \\ \text{where } \mu_f &= \frac{1}{\sqrt{\alpha_t}}z_t \quad \sigma_f = \sqrt{\frac{1-\alpha_t}{\alpha_t}} \quad \mu_g = \sqrt{\bar{\alpha}_{t-1}}x \quad \sigma_g = \sqrt{1-\bar{\alpha}_{t-1}} \\ \mu_{fg} &= \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})z_t + \sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)x}{1-\bar{\alpha}_t} \quad \sigma_{xy} = \frac{(1-\bar{\alpha}_{t-1})(1-\alpha_t)}{1-\bar{\alpha}_t} \quad S \text{ is independent of } z_{t-1} \end{aligned} \quad (84)$$

把上式两边对 z_{t-1} 求积分，可得

$$\begin{aligned} \frac{S}{K\sqrt{\alpha_{t-1}}} \int \frac{1}{\sqrt{2\pi\sigma_{fg}^2}} \exp\left\{\frac{-(z_{t-1} - \mu_{fg})^2}{2\sigma_{fg}^2}\right\} dz_{t-1} &= \int q(z_{t-1}|z_t, x) dz_{t-1} = 1 \\ \Rightarrow \quad \frac{S}{K\sqrt{\alpha_{t-1}}} &= 1 \end{aligned} \quad (85)$$

把上式的结果重新代入式(84)，可得

$$q(z_{t-1}|z_t, x) = \mathcal{N}(\mu_{fg}, \sigma_{fg}^2) = \mathcal{N}\left(\frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})z_t + \sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)x}{1-\bar{\alpha}_t}, \frac{(1-\bar{\alpha}_{t-1})(1-\alpha_t)}{1-\bar{\alpha}_t}\right) \quad (86)$$

简化 Lower Bound

将 $q(z_{1:T}|x)$ 的“正向展开”形式（式(75)）代入 $\mathcal{L}(\theta)$ ，可得

$$\begin{aligned} \mathcal{L}(\theta) &= \int q(z_{1:T}|x) \log \frac{p(x, z_{1:T})}{q(z_{1:T}|x)} dz_{1:T} \\ &= \int q(z_{1:T}|x) \log \frac{p(x|z_1)p(z_1|z_2)\cdots p(z_{T-1}|z_T)p(z_T)}{q(z_1|x)q(z_2|z_1)\cdots q(z_T|z_{T-1})} dz_{1:T} \end{aligned} \quad (87)$$

比较 $p(x, z_{1:T})$ 及 $q(z_{1:T}|x)$ 的形式，可以看出随机变量 $Z_{1:T}$ 存在的概率分布的形式不同，比如，在 p 模型中， Z_1 变量存在的条件概率分布依赖于 Z_2 （即 $q(z_1|z_2)$ ），而在 q 模型中， Z_1 存在的条件概率分布依赖于 X （即 $q(z_1|x)$ ），这样的形式不利于“凑全概率公式”及“凑 KL 散度”。

于是，将 $q(z_{1:T}|x)$ 的“条件展开”形式(式(79))代入 $\mathcal{L}(\theta)$ ，可得

$$\begin{aligned}
\mathcal{L}(\theta) &= \int q(z_{1:T}|x) \log \frac{p(x|z_1)p(z_1|z_2)p(z_2|z_3)\cdots p(z_{T-1}|z_T)p(z_T)}{q(z_1|z_2,x)q(z_2|z_3,x)\cdots q(z_{T-1}|z_T,x)q(z_T|x)} dz_{1:T} \\
&= \int q(z_{1:T}|x) \log p(x|z_1) dz_{1:T} - \sum_{t=2}^T \int q(z_{1:T}|x) \log \frac{q(z_{t-1}|z_t,x)}{p(z_{t-1}|z_t)} dz_{1:T} - \int q(z_{1:T}|x) \log \frac{q(z_T|x)}{p(z_T)} dz_{1:T} \\
&= \underbrace{\int q(z_1|x) \log p(x|z_1) dz_1}_{L_1} - \sum_{t=2}^T \underbrace{\int q(z_t|x) \overbrace{KL[q(z_{t-1}|z_t,x)\|p(z_{t-1}|z_t)]}^{KL_t} dz_t}_{L_t} - \underbrace{\overbrace{KL[q(z_T|x)\|p(z_T)]}^{KL_{T+1}}}_{L_{T+1}}
\end{aligned} \tag{88}$$

上述各项的简化利用了全概率公式。例如：

$$\int q(z_{1:T}|x) \log p(x|z_1) dz_{1:T} = \int \left\{ \int q(z_{1:T}|x) dz_{2:T} \right\} \log p(x|z_1) dz_1 = \int q(z_1|x) \log p(x|z_1) dz_1 \tag{89}$$

可以看出，通过此方式，能把大多数积分消除掉。

下面分析上述各项的特点：

- 起始项 L_1 是类似于 VAE 的重建项 (Reconstruction Term)。
- 结束项 L_{T+1} 是标准的 KL 散度，与 VAE 的一致项 (Consistent Term) 类似，但在 DDPM 中， q 是固定的模型， $p(z_T)$ 设置为标准高斯分布，所以此项是固定的，不包含参数，优化过程可忽略。另外，由于 $p(z_t|x)$ 的均值会随着 t 增大而减小，当 t 较大时， $p(z_t|x)$ 接近标准高斯分布，所以结束项的值会接近零，这一点跟 VAE 有较大的差别，VAE 中 KL 项的值一般会 Reconstruction Term 比较大，会导致 $p(x)$ 难以优化。
- 中间项 $\{L_t\}_{t=2}^T$ 是 KL 散度的期望，可以看作是 VAE 的一致项的改进版，因为 $p(z_{t-1}|z_t)$ 是条件概率，所以中间项考虑了所有可能的条件，然后根据概率计算其加权和。

3.3.3 三种优化(预测)方式

Predict Next Step

下面进一步确定各项的变换函数的表达式。

在 DDPM 模型， $p(z_t|z_{t+1})$ 的方差常设置为固定值，并与 $q(z_t|z_{t+1}, x)$ 的方差相同，从而方便 KL 散度的计算。所以，根据式(19)的结论，可得

$$\begin{aligned}
p(x|z_1) &= C_1 \|\mu_{\theta_1}(z_1) - x\|^2 \quad \text{where } C_1 = \frac{1}{2\sigma_1^2} \log \left(\sqrt{(2\pi)^k \sigma_1} \right) \\
KL_t &= C_t \|\mu_{\theta_t}(z_t) - m_t\|^2 \quad \text{where } C_t = \frac{1 - \bar{\alpha}_t}{2(1 - \bar{\alpha}_{t-1})(1 - \alpha_t)} \quad m_t = \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})z_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)x}{1 - \bar{\alpha}_t}
\end{aligned} \tag{90}$$

把式(90)代入式(88)，可得

$$\mathcal{L}(\theta) = \underbrace{C_1 \int q(z_1|x) \|\mu_{\theta_1}(z_1) - x\|^2 dz_1}_{L_1} - \sum_{t=1}^T \underbrace{C_t \int q(z_t|x) \|\mu_{\theta_t}(z_t) - m_t\|^2 dz_t}_{L_t} + const \tag{91}$$

至此，我们得到一个相对简单的目标函数。 $L_1, L_2 \dots L_T$ 各项均是随机变量的变换的期望，其概率分布 $q(z_t|x)$ 均是高斯分布，且无参数，所以各项均可通过 Monte Carlo 近似表示。由于优化目标是使 $\mu_{\theta_t}(z_t)$ 尽量接近 m_t ，而 m_t 是在 X 和 Z_t 已知条件下随机变量 z_{t-1} 的均值 ($q(z_{t-1}|z_t, x)$)，所以此优化方式也被为“Predict Next Step” ($z_{t-1} \leftarrow z_t$)。

Predict X

把 $\mu_{\theta_t}(z_t)$ 重新设置成如下形式：

$$\begin{aligned}\mu_{\theta_t}(z_t) &= \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})z_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)\hat{\mu}_{\theta_t}(z_t)}{1 - \bar{\alpha}_t} & t \in \{2, \dots, T\} \\ \mu_{\theta_1}(z_1) &= \hat{\mu}_{\theta_1}(z_1)\end{aligned}\quad (92)$$

其中， $\hat{\mu}_{\theta_t}(z_t)$ 是任意关于 z_t 的函数，可以是 CNN 或 Transformer 等。

将式(92)代入(90)，可得

$$KL_t = K_t \|\hat{\mu}_{\theta_t}(z_t) - x\|^2 \quad \text{where } K_t = \frac{\bar{\alpha}_{t-1}(1 - \alpha_t)}{2(1 - \bar{\alpha}_{t-1})(1 - \bar{\alpha}_t)} \quad t \in \{2, \dots, T\} \quad (93)$$

于是， $\mathcal{L}(\theta)$ 可写成

$$\mathcal{L}(\theta) = \underbrace{C_1 \int q(z_1|x) \|\hat{\mu}_{\theta_1}(z_1) - x\|^2 dz_1}_{L_1} - \sum_{t=2}^T \underbrace{K_t \int q(z_t|x) \|\hat{\mu}_{\theta_t}(z_t) - x\|^2 dz_t}_{L_t} \quad (94)$$

可以看出，把 $\mu_{\theta_t}(z_t)$ 的形式简单调整后，新的可学习函数 $\hat{\mu}_{\theta_t}(z_t)$ 的学习目标为 x ，即给定的训练数据，所以此优化方式被称之为“Predict $\mathbf{X}(x \leftarrow z_t)$ ”。

Predict Noise

下面推导第三种优化方式。

可以看出，式(88)各项均是随机变量的变换的期望。例如 L_t 项，随机变量为 Z_t ，概率分布为 $q(z_t|x) = \mathcal{N}(\sqrt{\bar{\alpha}_t}x, 1 - \bar{\alpha}_t)$ ，变换函数为 $KL[q(z_{t-1}|z_t, x)\|p(z_{t-1}|z_t)]$ 。根据 1.3 节的结论，可使用随机变量替换对各期望项进行转化。

构建一个临时随机变量 Y

$$Y = g_t(X, \epsilon) = \sqrt{\bar{\alpha}_t}X + (1 - \bar{\alpha}_t)\epsilon \quad \epsilon \sim \mathcal{N}(0, I) \triangleq \omega(\epsilon) \quad (95)$$

可以看出，在 X 已知条件下， Y 的概率分布 $\eta(y|x)$ 与 $q(z_t|x)$ 相同，所以下述两个期望值相等

$$\int q(z_t|x) KL[q(z_{t-1}|z_t, x)\|p(z_{t-1}|z_t)] dz_t = \int \eta(y|x) KL[q(z_{t-1}|y, x)\|p(z_{t-1}|y)] dy \quad (96)$$

同时，由于随机变量 Y 与 ϵ 存在变换关系 $g_t(X, \epsilon)$ ，应用 1.3 节的结论，可得

$$L_t = \int \eta(y|x) KL[q(z_{t-1}|y, x)\|p(z_{t-1}|y)] dy = \int \omega(\epsilon) KL[q(z_{t-1}|g(x, \epsilon), x)\|p(z_{t-1}|g(x, \epsilon))] d\epsilon \quad (97)$$

把式(90)中的 KL_t 代入上式，得

$$L_t = C_t \int \omega(\epsilon) \|\mu_{\theta_t}(\textcolor{red}{g_t(x, \epsilon)}) - \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})(\sqrt{\bar{\alpha}_t}x + (1 - \bar{\alpha}_t)\epsilon) + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)x}{1 - \bar{\alpha}_t}\|^2 d\epsilon \quad (98)$$

由于 $\mu_{\theta_t}(\textcolor{red}{g_t(x, \epsilon)})$ 是任意关于 $g_t(x, \epsilon)$ 的函数，所以可把 $\mu_{\theta_t}(\textcolor{red}{g_t(x, \epsilon)})$ 特意设计成如下形式

$$\mu_{\theta_t}(\textcolor{red}{g_t(x, \epsilon)}) = \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})(\sqrt{\bar{\alpha}_t}x + (1 - \bar{\alpha}_t)\overbrace{\bar{\mu}_{\theta_t}(\textcolor{red}{g_t(x, \epsilon)})}^{\textcolor{orange}{\hat{\mu}_{\theta_t}(\textcolor{red}{g_t(x, \epsilon)})}} + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)x}{1 - \bar{\alpha}_t} \quad (99)$$

其中， $\bar{\mu}_{\theta_t}(\textcolor{red}{g_t(x, \epsilon)})$ 是任意关于 $g_t(x, \epsilon)$ 的函数。把(99)代入(98)并应用 Monte Carlo 积分可得

$$\begin{aligned}L_t &= D_t \int \omega(\epsilon) \|\bar{\mu}_{\theta_t}(\textcolor{red}{g_t(x, \epsilon)}) - \epsilon\|^2 d\epsilon \quad \text{where } D_t = \frac{(1 - \bar{\alpha}_{t-1})(1 - \bar{\alpha}_t)\alpha_t}{2(1 - \alpha_t)} \\ &\approx D_t \sum_{j=0}^N \|\bar{\mu}_{\theta_t}(\textcolor{red}{g_t(x, \epsilon_j)}) - \epsilon_j\|^2 \quad \text{where } \epsilon_j \sim \omega(\epsilon)\end{aligned}\quad (100)$$

同理，可得到 L_1

$$L_1 = C_1 \int \omega(\epsilon) \|\bar{\mu}_{\theta_1}(\textcolor{red}{g}_1(\mathbf{x}, \epsilon)) - \mathbf{x}\|^2 d\epsilon \approx C_1 \sum_{j=0}^N \|\bar{\mu}_{\theta_1}(\textcolor{red}{g}_1(\mathbf{x}, \epsilon_j)) - \mathbf{x}\|^2 \quad (101)$$

where $\mu_{\theta_1}(\textcolor{red}{g}_1(\mathbf{x}, \epsilon_j)) = \bar{\mu}_{\theta_1}(\textcolor{red}{g}_1(\mathbf{x}, \epsilon_j))$ $C_1 = \frac{1}{2\sigma_1^2} \log \left(\sqrt{(2\pi)^k \sigma_1} \right)$ $\epsilon_j \sim \omega(\epsilon)$

L_1 是重建项，与其它优化方式基本一致。 $\{L_t\}_{t=2}^T$ 经过变换后有较大的差别。对于 L_t 项， ϵ_j 是采样得到的噪声样本， x 是给定的训练数据， $g_t(x, \epsilon_j)$ 是一个加噪的变换，优化目标函数使 $\bar{\mu}_{\theta_t}(\textcolor{red}{g}_t(\mathbf{x}, \epsilon_j))$ 接近于 ϵ_j 。所以此优化方式也称为”Predict Noise”。

3.3.4 优化 MLE

上面介绍了单个样本的目标函数。在此基础上，可进一步构建完整的 MLE 目标函数。根据 MLE(maximum Likelihood Estimation) 的定义，理想的目标函数 $Q_{mle}(\theta)$ 为

$$Q_{mle}(\theta) = \int p_{data}(x) \log p_\theta(x) dx = \int q(x) \log p_\theta(x) dx \quad (102)$$

在式(72)中假设 $q(x) = p_{data}(x)$ 。

由于 $q(x)$ 是未知的，只知道它的一批样本 $\{x_i\}_{i=1}^N$ ，所以对上式采用 Monte Carlo 近似，得

$$Q_{mle}(\theta) = \int q(x) \log p_\theta(x) dx \approx \sum_{i=0}^N \log p_\theta(x_i) \quad (103)$$

此目标函数即为常用的形式。

实际操作中，如果样本过多，可通过 Mini Batch 的方式近似优化。

在学习到 $p_\theta(x)$ 的参数后，联合概率分布 $p_\theta(x, z_{1:T})$ 已经确定，则可通过 Ancestral Sampling(2.2节) 的方式，采样得到 X 的新样本。

把新样本 X_i 代入 $\mathcal{L}(\theta)$ ，可计算此样本的概率 $p_\theta(x)$ 的 Lower Bound。

3.4 DPM 模型的进一步分析

3.4.1 概括重要的结论

为了方便捋清思路，把一些重要结论重列一下。

待优化的 p 概率模型的形式 (性质 1)

$$p(x, z_{1:T}) = p(x|z_1) \prod_{t=2}^T p(z_{t-1}|z_t) p(z_T) = p(x|z_1)p(z_1|z_2)\dots p(z_{T-1}|z_T)p(z_T) \quad (104)$$

q 概率模型的“正向展开” (性质 2)

$$q(x, z_{1:T}) = q(x)q(z_1|x) \prod_{t=2}^T q(z_t|z_{t-1}) = q(x)q(z_1|x)q(z_2|z_1)\dots q(z_T|z_{T-1}) \quad (105)$$

q 概率模型的“反向展开” (性质 3)

$$q(x, z_{1:T}) = q(x|z_1) \prod_{t=2}^T q(z_{t-1}|z_t) q(z_T) = q(x|z_1)q(z_1|z_2)\dots q(z_{T-1}|z_T)q(z_T) \quad (106)$$

q 概率模型的“条件展开” (性质 4)

$$q(x, z_{1:T}) = \prod_{t=2}^T q(z_{t-1}|z_t) q(z_T|x)q(x) = q(z_1|z_2, x)\dots q(z_{T-1}|z_T, x)q(z_T|x)q(x) \quad (107)$$

概率分布 $q(z_t|z_{t-1})$ 有如下的形式 (性质 1)

$$q(z_t|z_{t-1}) = \mathcal{N}(\sqrt{\alpha_t}z_{t-1}, 1 - \alpha_t) \quad (108)$$

概率分布 $q(z_t|x)$ 有如下形式 (性质 5)

$$q(z_t|x) = \mathcal{N}(\sqrt{\bar{\alpha}_t}x, 1 - \bar{\alpha}_t) \quad \text{where } \bar{\alpha}_t \triangleq \prod_{i=1}^t \alpha_i \quad (109)$$

概率分布 $q(z_{t-1}|z_t, x)$ 有如下形式 (性质 6)

$$q(z_{t-1}|z_t, x) = \mathcal{N}\left(\frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})z_t + \sqrt{\bar{\alpha}_{t-1}}(1 - \alpha_t)x}{1 - \bar{\alpha}_t}, \frac{(1 - \bar{\alpha}_{t-1})(1 - \alpha_t)}{1 - \bar{\alpha}_t}\right) \quad (110)$$

$p(x)$ 存在如下的关系

$$\log p(x) = \underbrace{\int q(z_{1:T}|x) \log \frac{p(x, z_{1:T})}{q(z_{1:T}|x)} dz_{1:T}}_{\mathcal{L}(\theta)} + \underbrace{\int q(z_{1:T}|x) \log \frac{q(z_{1:T}|x)}{p(z_{1:T}|x)} dz_{1:T}}_{KL(q||p)} \quad (111)$$

$$\log p(x) \geq \mathcal{L}(\theta) \quad (112)$$

$\mathcal{L}(\theta)$ 可展开成如下的形式

$$\mathcal{L}(\theta) = \underbrace{\int q(z_1|x) \log p(x|z_1) dz_1}_{L_1} - \sum_{t=2}^T \underbrace{\int q(z_t|x) \overbrace{KL[q(z_{t-1}|z_t, x) \| p(z_{t-1}|z_t)]}^{KL_t} dz_t}_{L_t} - \underbrace{\overbrace{KL[q(z_T|x) \| p(z_T)]}^{KL_{T+1}}}_{L_{T+1}} \quad (113)$$

3.4.2 进一步理解目标函数

根据 MLE 目标函数的定义可知

$$Q_{mle}(\theta) = \int q(x) \log p_\theta(x) dx \quad \text{where } q(x) \triangleq p_{data}(x) \quad (114)$$

把式(111)代入上式得

$$Q_{mle}(\theta) \geq \int q(x) \mathcal{L}(\theta) dx \quad (115)$$

定义 $Q_{lb}(\theta) \triangleq \int q(x) \mathcal{L}(\theta) dx$, 并对 $\mathcal{L}(\theta)$ 展开, 可得

$$Q_{lb}(\theta) = \underbrace{\int q(x) \underbrace{\int q(z_1|x) \log p(x|z_1) dz_1 dx}_{L_1}}_{Q_1} - \sum_{t=2}^T \underbrace{\int q(x) \underbrace{\int q(z_t|x) \overbrace{KL[q(z_{t-1}|z_t, x) \| p(z_{t-1}|z_t)]}^{KL_t} dz_t dx}_{L_t}}_{Q_t} - \underbrace{\int q(x) \overbrace{KL[q(z_T|x) \| p(z_T)]}^{KL_{T+1}} dx}_{Q_{T+1}} \quad (116)$$

下面分析起始项、中间项和结束项的特点。

中间项

利用 Bayes 公式, 可把中间项 Q_t 重写为

$$\begin{aligned} Q_t &= \iint q(z_t, x) \overbrace{KL[q(z_{t-1}|z_t, x) \| p(z_{t-1}|z_t)]}^{KL_t} dz_t dx \\ &= \int q(z_t) \underbrace{\int q(x|z_t) \overbrace{KL[q(z_{t-1}|z_t, x) \| p(z_{t-1}|z_t)]}^{KL_t} dx}_{dz_t} dz_t \end{aligned} \quad (117)$$

由于最小化 $KL[q(z_{t-1}|z_t, x) \| p(z_{t-1}|z_t)]$ 等效于最小化 $KL[p(z_{t-1}|z_t) \| q(z_{t-1}|z_t, x)]$

$$\min_{\theta} KL[q(z_{t-1}|z_t, x) \| p(z_{t-1}|z_t)] \iff \min_{\theta} KL[p(z_{t-1}|z_t) \| q(z_{t-1}|z_t, x)] \quad (118)$$

所以，最小化 Q_t 等效于最小化 \tilde{Q}_t

$$\min_{\theta} Q_t \iff \min_{\theta} \tilde{Q}_t \quad \text{where } \tilde{Q}_t = \int q(z_t) \int q(x|z_t) KL[p(z_{t-1}|z_t) \| q(z_{t-1}|z_t, x)] dx dz_t \quad (119)$$

于是，当 Q_t 趋向于 0 时， \tilde{Q}_t 也会趋向于 0。

根据 1.12 节关于 KL 散度上限的结论， \tilde{Q}_t 存在如下的关系。

$$\begin{aligned} \tilde{Q}_t &= \int q(z_t) \underbrace{\int q(x|z_t) KL[p(z_{t-1}|z_t) \| q(z_{t-1}|x, z_t)] dx}_{dz_t} \\ &\geq \int q(z_t) KL[p(z_{t-1}|z_t) \| q(z_{t-1}|z_t)] dz_t \end{aligned} \quad (120)$$

于是，当 \tilde{Q}_t 趋向于 0 时， $KL[p(z_{t-1}|z_t) \| q(z_{t-1}|z_t)]$ 会趋向于 0， $p(z_{t-1}|z_t)$ 会接近于 $q(z_{t-1}|z_t)$ 。

综合上述两个关系，当 Q_t 趋向于 0 时， $p(z_{t-1}|z_t)$ 会接近于 $q(z_{t-1}|z_t)$ 。

因此，优化中间项，本质是让 $p(z_{t-1}|z_t)$ 去拟合 $q(z_{t-1}|z_t)$ 。

起始项

应用 Bayes 公式，起始项可做如下的变换

$$\begin{aligned} Q_1 &= \iint q(x) q(z_1|x) \log p(x|z_1) dx dz_1 \\ &= \iint q(z_1) q(x|z_1) \log p(x|z_1) dx dz_1 \\ &= - \iint q(z_1) q(x|z_1) \left(\log \frac{q(x|z_1)}{p(x|z_1)} - \log q(x|z_1) \right) dx dz_1 \\ &= - \int q(z_1) KL[q(x|z_1) \| p(x|z_1)] + \iint q(x, z_1) \log q(x|z_1) dx dz_1 \\ &= - \int q(z_1) KL[q(x|z_1) \| p(x|z_1)] + const \end{aligned} \quad (121)$$

其中，由于 q 是固定的模型，所以 $\iint q(x, z_1) \log q(x|z_1) dx dz_1$ 是不依赖于参数 θ 的常数。

可以看出，当最大化 Q_1 ，使 Q_1 变大时， $KL[q(x|z_1) \| p(x|z_1)]$ 会趋向于 0， $p(x|z_1)$ 会趋向于 $q(x|z_1)$ 。

因此，优化起始项，本质是让 $p(x|z_1)$ 去拟合 $q(x|z_1)$ 。

结束项

对于结束项， $p(x_T)$ 设置为固定的标准高斯分布。当 T 较大时， $q(z_T|x)$ 也接近于标准高斯分布，且接近于不依赖 X ，即

$$q(z_T|x) \approx q(z_T) \approx \mathcal{N}(0, I) \quad (122)$$

所以 $p(x_T)$ 与 $q(z_T|x)$ 非常相似， Q_{T+1} 项的值接近于 0。

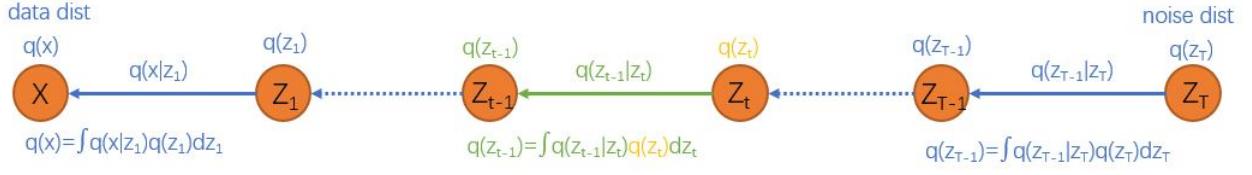


图 14: from noise dist to data dist

3.4.3 理解噪声分布向数据分布转变的过程

如图 14 所示, 根据 Bayes 公式, 可从反向递推计算每个节点的概率分布

$$\begin{aligned}
 q(z_{T-1}) &= \int q(z_{T-1}|z_T)q(z_T)dz_T \\
 &\dots \\
 q(z_{t-1}) &= \int q(z_{t-1}|z_t)q(z_t)dz_t \\
 &\dots \\
 q(x) &= \int q(x|z_1)q(z_1)dz_1
 \end{aligned} \tag{123}$$

在性质 1 中提到, 在设计 q 的参数 α_t 时, 会特意使 $q(z_T)$ 接近于 $\mathcal{N}(0, I)$, 所以 $q(z_T)$ 是已知的噪声分布。而 $q(x|z_1)$ 和 $\{q(z_{t-1}|z_t)\}_{t=2}^T$ 均是未知的概率分布。(提示, 根据 q 模型的假设, $q(z_1|x)$ 和 $\{q(z_t|z_{t-1})\}_{t=2}^T$ 是有解析形式的 (性质 1), $\{q(z_t|x)\}_{t=2}^T$ 是有解析形式的, $\{q(z_{t-1}|z_t, x)\}_{t=2}^T$ 是有解析形式的 (性质 6)。)

可以看出, $q(z_{t-1}|z_t)$ 起到一个“变换算子”作用, 将概率分布 $q(z_t)$ 映射成 $q(z_{t-1})$ 。这个变换过程类似于前面提到“高斯分布加权和”, 只不过它的基函数有复杂的形式。依次执行变换, 最终将变换至 $q(x)$ 。所以, 如果知道了“变换算子”, 也将能确定 $q(x)$ 。为了表述方便, 将 $q(z_t|z_{t-1})$ 称为“逆变换”算子, $q(z_{t-1}|z_t)$ 称为“正变换”算子。

由上一节分析可知, $q(z_{t-1}|z_t)$ 正是 $p(z_{t-1}|z_t)$ 拟合的目标。所以, 也可以这么理解, DPM 模型的训练是在学习“一串逆变换”, 这些变换依次顺序作用, 使一个噪声分布向数据分布 $q(x)$ 转变。

在理想情况下, 起始项 $p(x|z_1)$ 拟合至 $q(x|z_1)$, 中间项 $p(z_{t-1}|z_t)$ 拟合至 $q(z_{t-1}|z_t)$, 结束项 $p(x_T)$ 约等于 $q(z_T)$, 此时, $p(x)$ 也将拟合至 $q(x)$ 。

这告诉我们一个结论, 虽然目标函数只是 MLE 的下限, 并且 $q(z_{1:T}|x)$ 只是一个假设的分布, 但在理想情况下, $p(x)$ 还是能够拟合至真正的概率分布 $p_{data}(x)$ 。

此状态下联合概率分布 $p(x, z_{1:T})$ 为

$$p(x, z_{1:T}) = q(x|z_1) \prod_{t=2}^T q(z_{t-1}|z_t) q(z_T) \tag{124}$$

与式(106)比较, 可以看出, $p(x, z_{1:T})$ 与 $q(x, z_{1:T})$ 的“反向展开”完全一样。所以, 也可以这么认为, 优化 $Q_{lb}(\theta)$ 的是让 p 模型去学习 q 模型的“反向展开”。

3.4.4 $q(z_{t-1}|z_t)$ 概率分布的特点

概率分布 $q(z_{t-1}|z_t)$ 与 $q(x)$ 有关, 没有闭合的形式, 但存在一些方法, 可推测其形状, 并分析影响其形状的因素。

根据 Bayes 公式, 有

$$q(z_{t-1}|z_t) = \frac{q(z_t|z_{t-1})q(z_{t-1})}{q(z_t)} \tag{125}$$

当 z_t 是取固定值时, $q(z_t)$ 是常数, 所以 $q(z_{t-1}|z_t)$ 的形状只与 $q(z_t|z_{t-1})q(z_{t-1})$ 有关。

$$q(z_{t-1}|z_t) \propto q(z_t|z_{t-1})q(z_{t-1}) \quad \text{where } z_t \text{ is fixed} \tag{126}$$

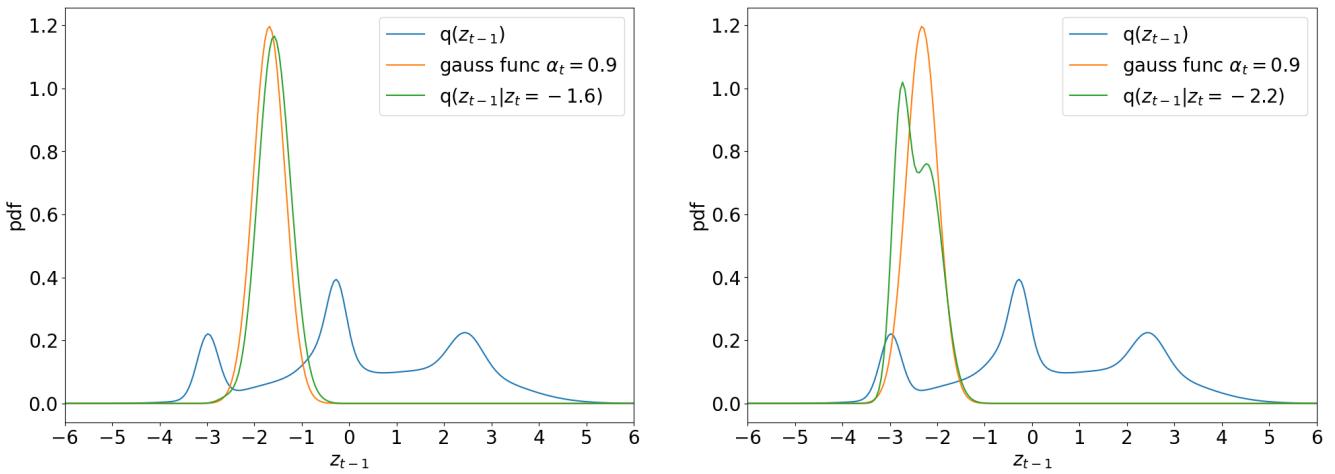


图 15:

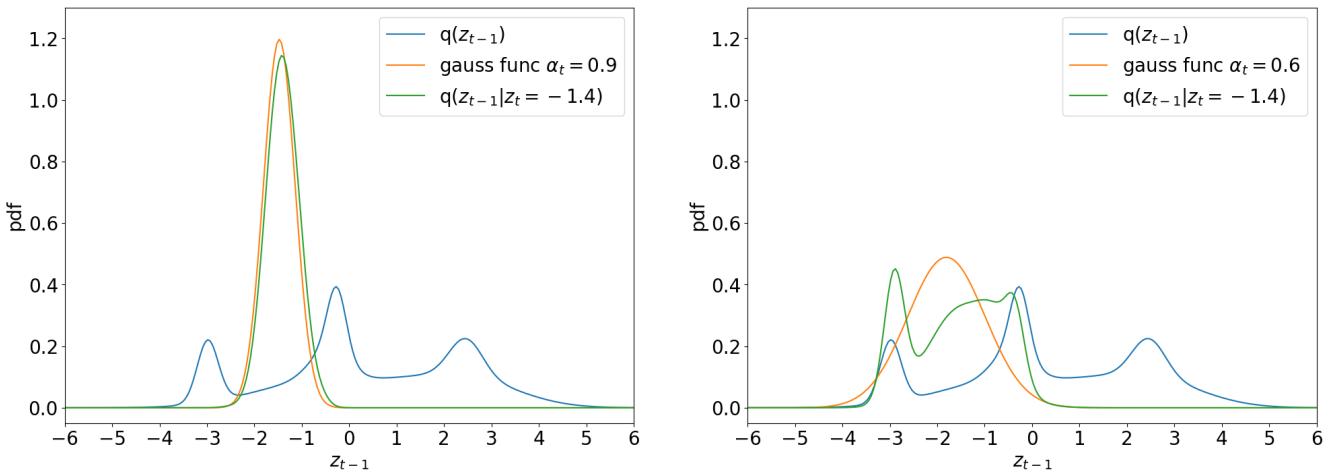


图 16:

由性质 1 可知, $q(z_t|z_{t-1})$ 为高斯分布, 于是有

$$\begin{aligned} q(z_{t-1}|z_t) &\propto \frac{1}{\sqrt{2\pi}(1-\alpha_t)} \exp \frac{-(z_t - \sqrt{\alpha_t}z_{t-1})^2}{2(1-\alpha_t)} q(z_{t-1}) \quad \text{where } z_t \text{ is fixed} \\ &= \underbrace{\frac{1}{\sqrt{\alpha_t}} \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-(z_{t-1} - \mu)^2}{2\sigma^2}}_{GF} q(z_{t-1}) \quad \text{where } \mu = \frac{z_t}{\sqrt{\alpha_t}} \quad \sigma = \sqrt{\frac{1-\alpha_t}{\alpha_t}} \end{aligned} \quad (127)$$

可以看出, GF 部分是关于 z_{t-1} 的高斯函数, 均值为 $\frac{z_t}{\sqrt{\alpha_t}}$, 方差为 $\sqrt{\frac{1-\alpha_t}{\alpha_t}}$ 。所以 $q(z_{t-1}|z_t)$ 的形状由“高斯函数与 $q(z_{t-1})$ 相乘”决定。此函数形状有几个特点:

- 有效区域的范围基本小于高斯函数的有效区域 (3 倍方差), 除非 $q(z_{t-1})$ 存在剧烈的波动。
- 在有效区域内, 当 $q(z_{t-1})$ 波动较小时, $q(z_{t-1}|z_t)$ 的形状基本是高斯形状; 当 $q(z_{t-1})$ 波动较大时, $q(z_{t-1}|z_t)$ 的形状将不再是高斯, 并且会跟随 $q(z_{t-1})$ 出现较大的波动。具体可看图 15, 其是一个一维的例子。
- 当 α_t 较小时, 高斯函数的方差也将较大, 有效区域的范围也将较大, 此时, 包含波动变化的 $q(z_{t-1})$ 的可能性也将变大, 所以 $q(z_{t-1}|z_t)$ 出现非高斯形状的可能性也变大。反之, 当 α_t 较大时, 有效区域较小, $q(z_{t-1}|z_t)$ 将更可能像高斯形状。具体可看图 16, 其是一个一维例子, 同样的 z_t , 不同的 α_t 值, 导致明显不同的 $q(z_{t-1}|z_t)$ 。
- 当 α_t 较大或 $q(z_{t-1})$ 波动变化较缓时, $q(z_{t-1}|z_t)$ 的均值与 z_t 有较明显的线性关系。

总结一下, $q(z_{t-1}|z_t)$ 的形状总体上是“局部性”的, “是否像高斯”与 α_t 的值和 $q(z_{t-1})$ 本身的复杂程度有关。由于 $p(z_{t-1}|z_t)$ 的函数形式假设为“条件高斯”, “非高斯形状”将影响 $p(z_{t-1}|z_t)$ 拟合 $q(z_{t-1}|z_t)$ 。

由于 $q(z_{t-1}|z_t)$ 与 α_t 的值和 $q(z_{t-1})$ 的函数形式有关, 所以, 可以这么认为, $q(z_{t-1}|z_t)$ 包含了 $q(z_{t-1})$ 的部分信息。 α_t 的值越小(方差越大), 包含的信息越多; α_t 的值越大(方差越小), 包含的信息越少。特别地, 当 $\alpha_t \rightarrow 1$ 时, Z_{t-1} 和 Z_t 之间是确定的关系, 不包含随机性, $q(z_{t-1}|z_t)$ 不包含 $q(z_{t-1})$ 的信息; 当 $\alpha_t \rightarrow 0$ 时, $q(z_{t-1}|z_t)$ 包含 $q(z_{t-1})$ 的全部信息, 此时, Z_{t-1} 和 Z_t 相互独立。

下面看两个完整的“变换”例子(图 17 至图 19)。一个一维的随机变量, 经过多次的变换, 其概率分布最终转变成近似的标准高斯分布。图 17 展示的是 $\alpha_t = 0.6$ 的转换过程, 可以看出, 转换速度较快, $q(z_5)$ 基本是近似的标准高斯分布; 图 19 展示的是对应的 $q(z_{t-1}|z_t)$, 可以看出 $q(z_{t-1}|z_t)$ 总体偏胖, 方差较大, 并且函数形式较复杂, 特别是前两个转换过程。在前两个转换过程, 由于 $q(z_{t-1})$ 波动变化较大, 所以, 对于大多的 z_t , $q(z_{t-1}|z_t)$ 不再是高斯形状, 并且均值与 z_t 不再有较简单的线性关系。图 18 展示的是 $\alpha_t = 0.9$ 的转换过程, 可以看出, 转换速度较慢, $q(z_9)$ 才基本是标准高斯分布; 图 20 展示的是对应的 $q(z_{t-1}|z_t)$, 可以看出 $q(z_{t-1}|z_t)$ 总体偏瘦, 方差较小, 并且函数形式相对较简单, 特别是前两个转换过程。

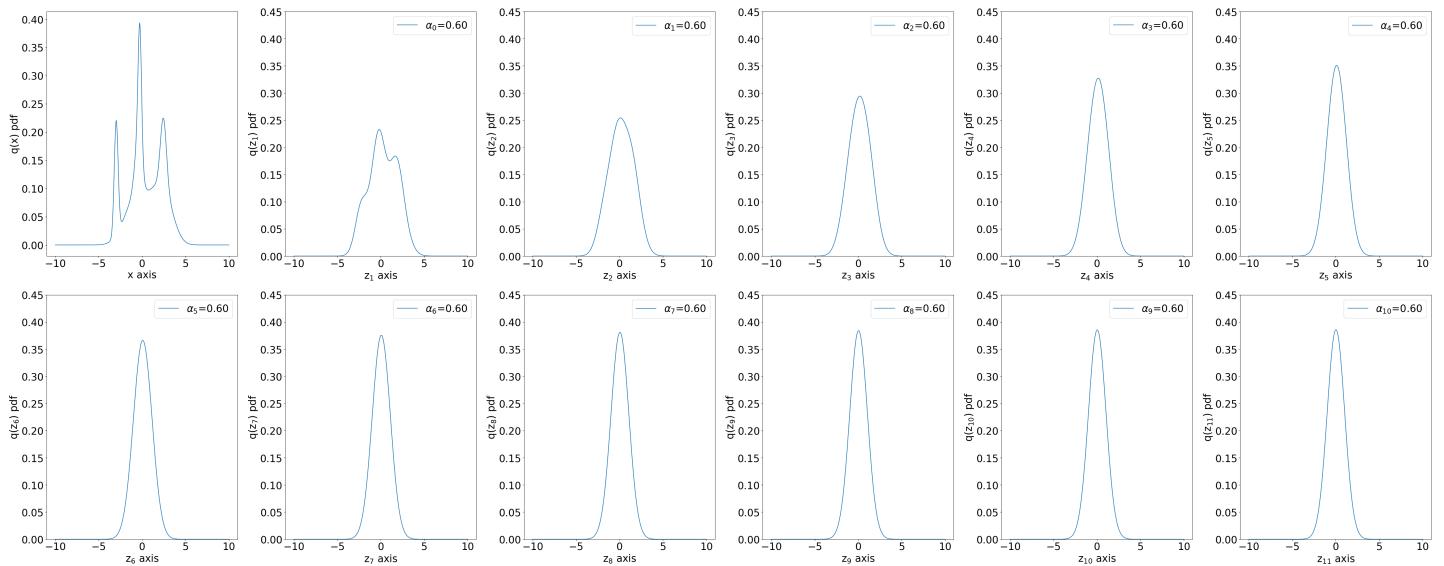


图 17: $q(z_t)$ distribution when $\alpha_t = 0.6$

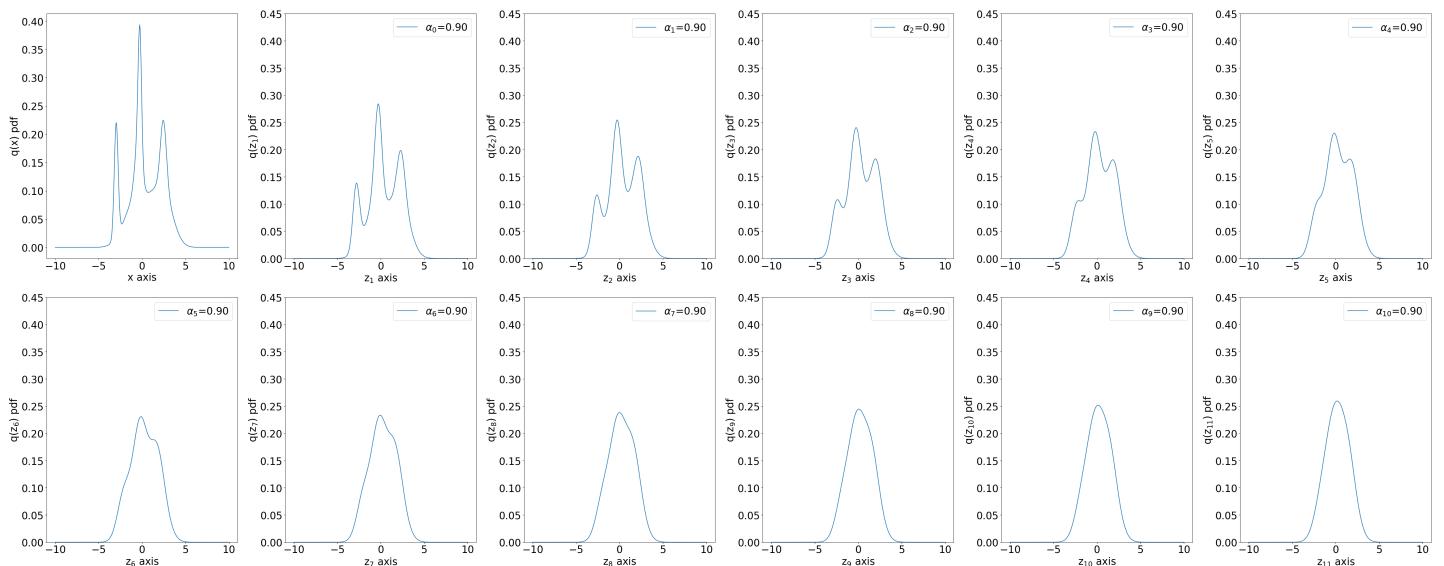


图 18: $q(z_t)$ distribution when $\alpha_t = 0.9$

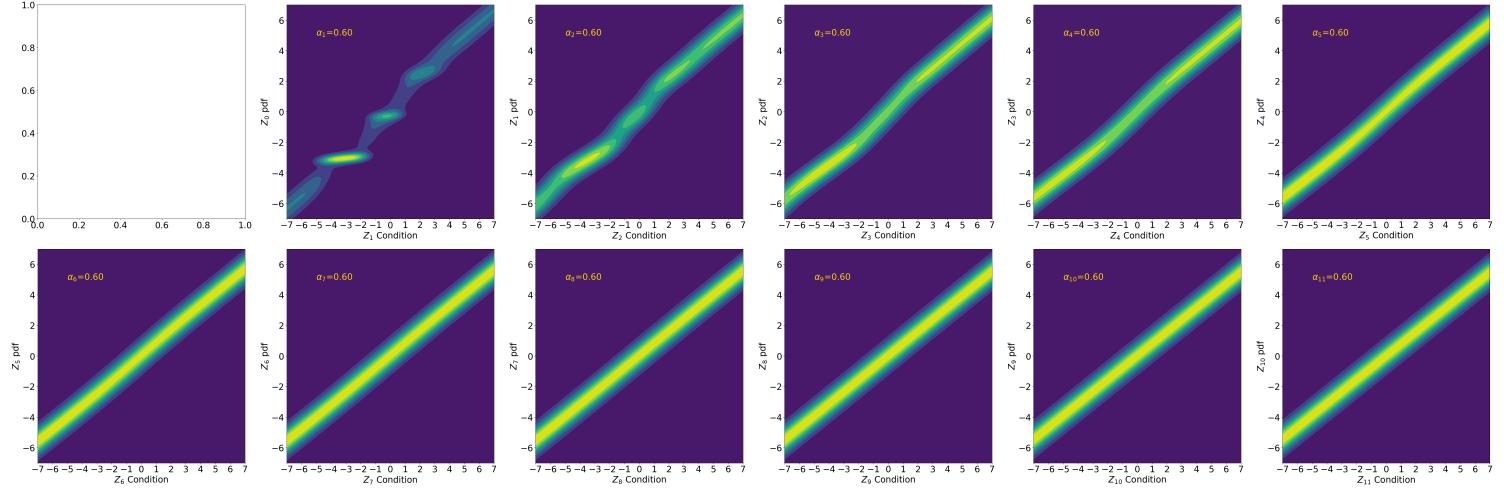


图 19: $q(z_{t-1}|z_t)$ distribution when $\alpha_t = 0.6$

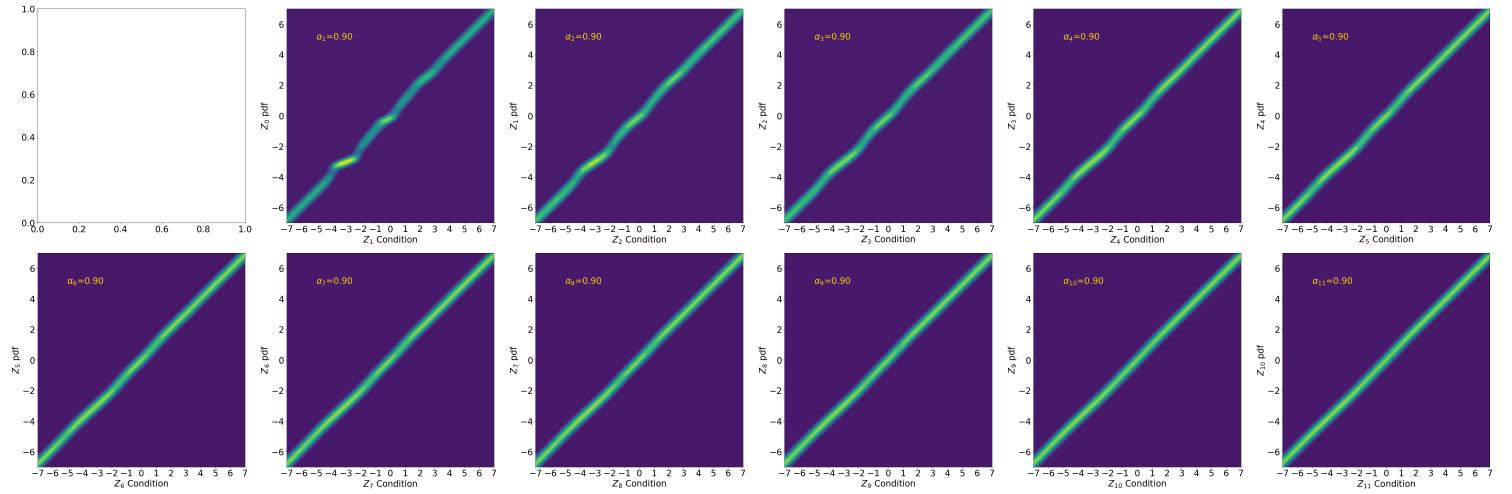


图 20: $q(z_{t-1}|z_t)$ distribution when $\alpha_t = 0.9$

3.4.5 $q(z_{t-1}|z_t)$ 逆变换的输入敏感度

由上面的分析可知, $q(x)$ 需要经过一系列变换才能得到, 由于各个变换之间是“串联”的关系, 总误差与 $q(z_T)$ 的输入误差有着指数的关系。当变换使误差减少时, 总误差将随变换的次数指数级地衰减; 当变换使误差增大时, 总误差将随变换的次数指数级地放大。所以, 有必要分析各个变换的输入敏感度。

经过实验总结分析可知, $q(z_{t-1}|z_t)$ 逆变换有着一定的抗噪能力, 输出的误差不大于输入的误差。具体特点如下:

- 去噪能力与 α_t 的值有着反比的关系。 α_t 越小 (方差越大), 去噪能力越强。 α_t 越大 (方差越小), 去噪能力越弱。具体可看图 21 的例子, $q(x)$ 一个初始的分布, 经过两个不同的变换后得到 $q(z_1)$, 在 $q(z_1)$ 在加上类似的噪声, 加噪后与原分布的 JS 散度均 0.20。经过 $\alpha_t = 0.90$ 的逆变换后, 带噪的 $q(x)$ 与原始的 $q(x)$ 的 JS 散度为 0.02。经过 $\alpha_t = 0.98$ 的逆变换后, 带噪的 $q(x)$ 与原始的 $q(x)$ 的 JS 散度为 0.12。带噪的概率分布曲线为金黄色, 原始的概率分布曲线为蓝色。
- 一个强的去噪逆变换 (较小的 α_l 值) 可由多个弱的去噪逆变换 (多个较大的 $\{\alpha_j\}_{j=0}^K$ 值) 等效代替, 其中 $\alpha_l = 1 - \sum_{j=0}^K (1 - \alpha_j)$ 。具体可看图 22 的例子, $q(x)$ 一个初始的分布, 分别经过两个不同系列的变换。第一个系列变换 (第一行) 包含 5 个子变换, 每个子变换对应的 α_t 均为 0.98; 第二个系列变换 (第二行) 只

包含一个变换，对应的 α_t 均为 0.90。分别对末尾的概率分布加上类似的噪声，使 JS 散度均为 0.19。分别使用逆变换，得到带噪的 $q(x)$ 信号，与原始信号相比，JS 散度均为 0.05。

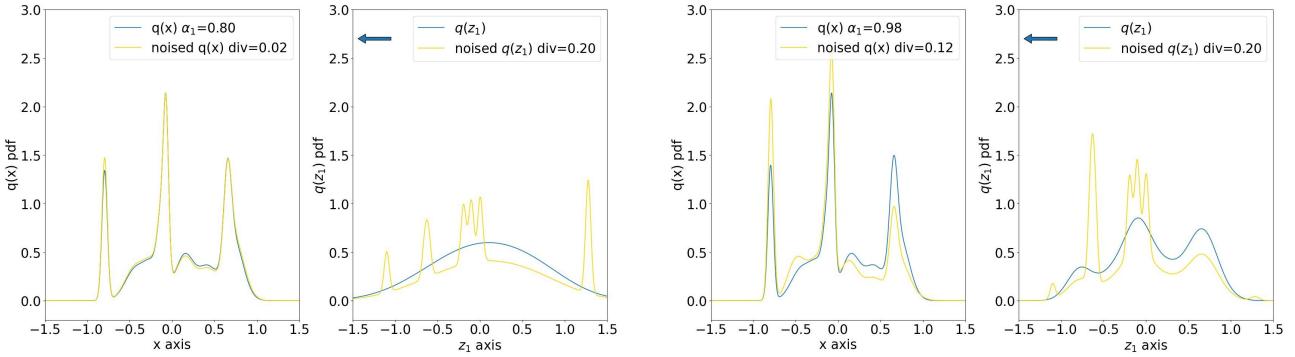


图 21: Anti-noise capability with different α_t

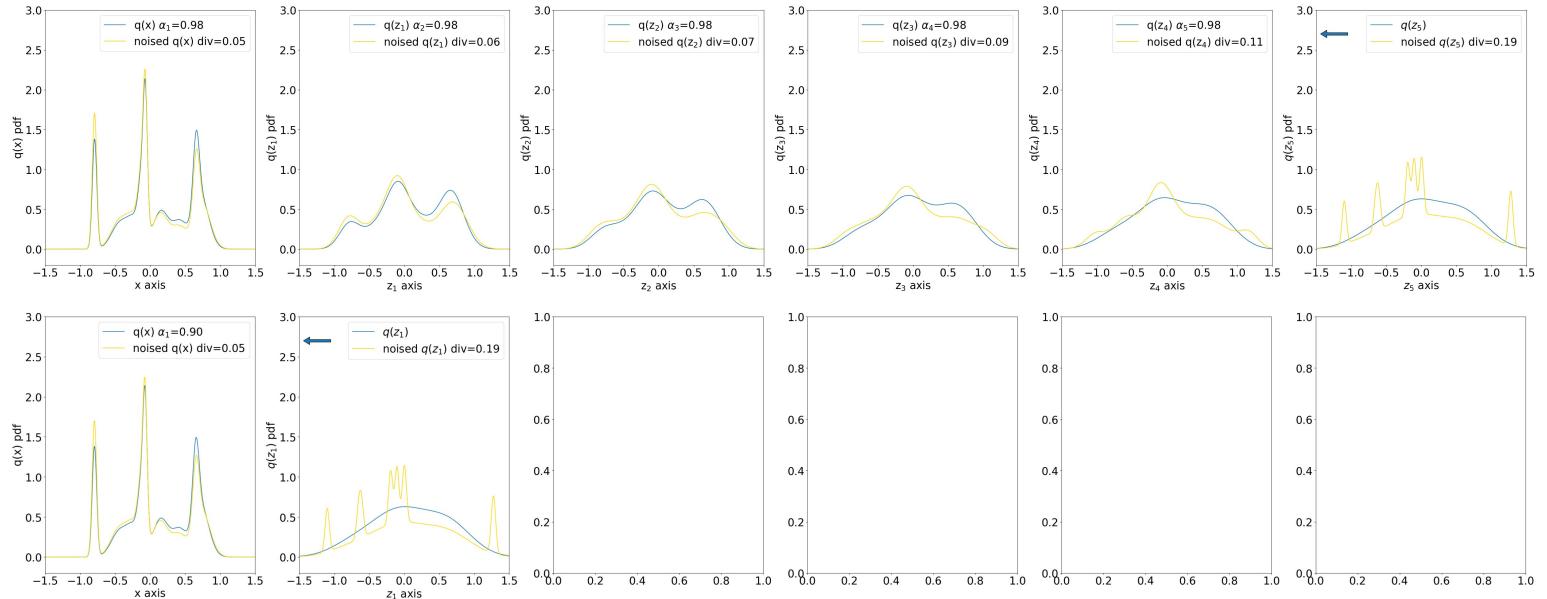


图 22: Multiple small trans is equivalent to one big trans

关于 $q(z_{t-1}|z_t)$ 逆变换具备抗噪能力的原因，可从几个角度理解。

较简单的解释。逆变换依赖于 $q(z_{t-1}|z_t)$ 和 $q(z_t)$ 两部分的信息，由于 $q(z_{t-1}|z_t)$ 已经包含着部分 $q(z_{t-1})$ 的信息，所以 $q(z_{t-1})$ 只部分依赖 $q(z_t)$ ，因此， $q(z_t)$ 的噪声只会部分影响 $q(z_{t-1})$ 。

较复杂的解释。逆变换的过程可理解为“线性加权和”，基函数为不同 z_t 对应的 $q(z_{t-1}|z_t)$ ，加权系数为 $q(z_t)$ 。当 α_t 小于 1 时，基函数对应的高斯函数方差非零，基函数包含着部分 $q(z_{t-1})$ 信息，并且各个基函数之间存在着部分重叠，因此基函数之间存在一定的相关性，并非相互独立，所以基函数所张成 (span) 的空间小于 $q(z_t)$ 所在的空间。同时，考虑到逆变换是一个线性的变换，所以，逆变换可认为是一个压缩映射 (以 JS Divergence 为 Metric 的 Contraction Mapping)。压缩变换表明，变换后的带噪信号与原始信号的距离，小于变换前的带噪信号与原始信号的距离，因此，逆变换具备一定的抗噪能力。

上述关于 “ $q(z_{t-1}|z_t)$ 是一个 Contraction Mapping” 的结论并没有严谨的证明，只是一个直观的解释，但大量的模拟实验支持此结论。

通过图 23 和 24 的例子可直观地理解 “ $q(z_{t-1}|z_t)$ 是一个压缩映射”。第 1 子图展示了 $q(z_{t-1})$ ，其是一个随机初始化的概率分布，经过 $q(z_t|z_{t-1})$ 变换后得到 $q(z_t)$ (第 2 子图) 及相应的逆变换 $q(z_{t-1}|z_t)$ (第 2 子图)。第 4 子图随机初始化两个新的 $q(z_t)$ 的分布，分布之间的 JS 散度为 0.41；第 5 子图展示变换后得到的 $q(z_{t-1})$ ，JS 散度为 0.12。从图像曲线和度量值都可以看出，变换后分布之间的距离变小。类似的情况可在剩余的配对中观

察得到。

分析两个极端的情况可以佐证“ $q(z_{t-1}|z_t)$ 是一个压缩映射”。

当 $\alpha_t \rightarrow 1$ 时，基函数对应的高斯函数的方差趋向于无穷小，因此基函数几乎不包含 $q(z_{t-1})$ 的信息，并且收缩成 Dirac delta 函数。此状态的基函数之间相互正交，线性无关， z_t 与 z_{t-1} 之间的关系变成确定的线性关系，因此基函数所张成 (span) 的空间等于 $q(z_t)$ 所在的空间，所以逆变换为保距映射。保距映射说明输入噪声等幅度传递至输出。

当 $\alpha_t \rightarrow 0$ 时，基函数对应的高斯函数的方差趋向于无穷大，因此基函数几乎包含完整的 $q(z_{t-1})$ 信息，接近于 $q(z_{t-1})$ 。此状态的基函数几乎一模一样，相关性极强，基函数所张成 (span) 的空间非常非常小。在这种情况下，不管加权系数 $q(z_t)$ 有多么明显的扰动， $q(z_{t-1})$ 将基本保持不变。所以，此状态的逆变换是一个压缩率极大的压缩映射。不管 $q(z_t)$ 包含多大的噪声，输出都将在 $q(z_{t-1})$ 附近。

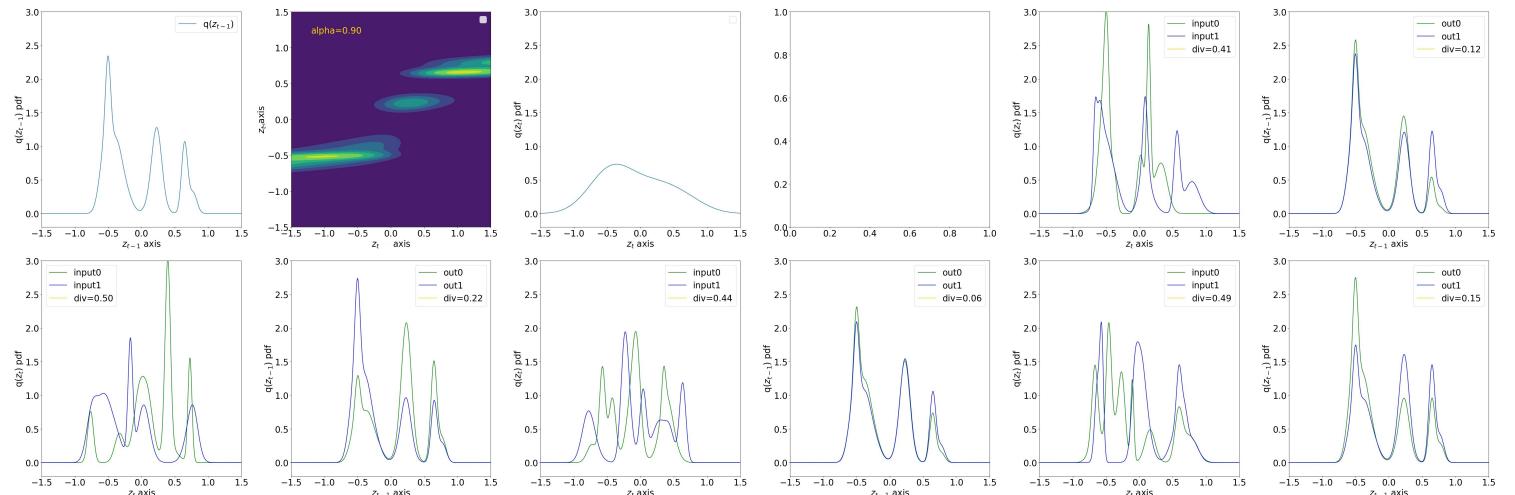


图 23: Contraction mapping with $\alpha=0.90$

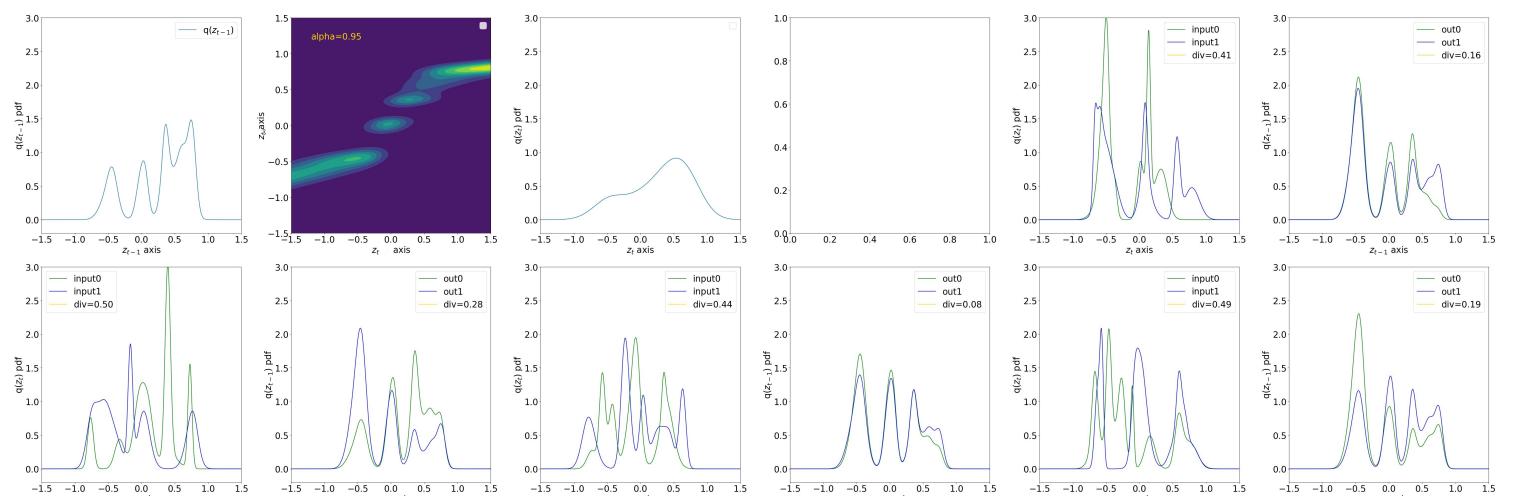


图 24: Contraction mapping with $\alpha=0.95$

3.4.6 $p(z_{t-1}|z_t)$ 拟合误差对逆变换的影响

由上面的分析可知，在 q 模型中， $q(z_{t-1}|z_t)$ 是未知的，需要通过 $p(z_{t-1}|z_t)$ 拟合近似。由于 $p(z_{t-1}|z_t)$ 设置为条件高斯的形式，条件高斯的复杂度不高，所以会造成一定的拟合误差。由于逆变换是以 $q(z_{t-1}|z_t)$ 为基函数的线性加权和，当基函数的存在限制时，将会影响 $q(z_{t-1})$ 分布的恢复。

拟合误差对逆变换结果的影响会有两个特点。

- 当使用高斯函数拟合较复杂的基函数时，基函数的细节将有所丢失，从而进一步导致 $q(z_{t-1})$ 函数细节的损失。图 25 是一个例子。第一行显示的各个随机变量的概率分布 $q(x)$ 及 $\{q(z_i)\}_{i=1}^3$ ；第二行显示的是基函数 $q(x|z_1)$ ；第三行显示的是用条件高斯拟合得到的基函数 $p(x|z_1)$ 。第二行首图展示了三个不同 z_1 值对应的 $q(x|z_1)$ 概率分布，第三行的首图展示了高斯拟合后的结果。可以看出，由于基函数过于复杂，拟合后的基函数与原始的函数有较大的差别。相应地，在第一行的首图可以看到，恢复后的 $q(x)$ (金黄色曲线) 细节明显减少，曲线更加光滑。
- 当 α_t 较大或者 $q(z_{t-1})$ 波动变化较缓时，基函数 $q(z_{t-1}|z_t)$ 与高斯函数较相似，此状态的拟合误差较小，对 $q(z_{t-1})$ 的恢复影响也将较小。图 26 是一个与图 25 相似的例子，只不过其使用较大的 α 值。可以看出，基函数 $q(x|z_1)$ 能得到较好的拟合，恢复后的 $q(x)$ (金黄色曲线) 细节丢失较少。

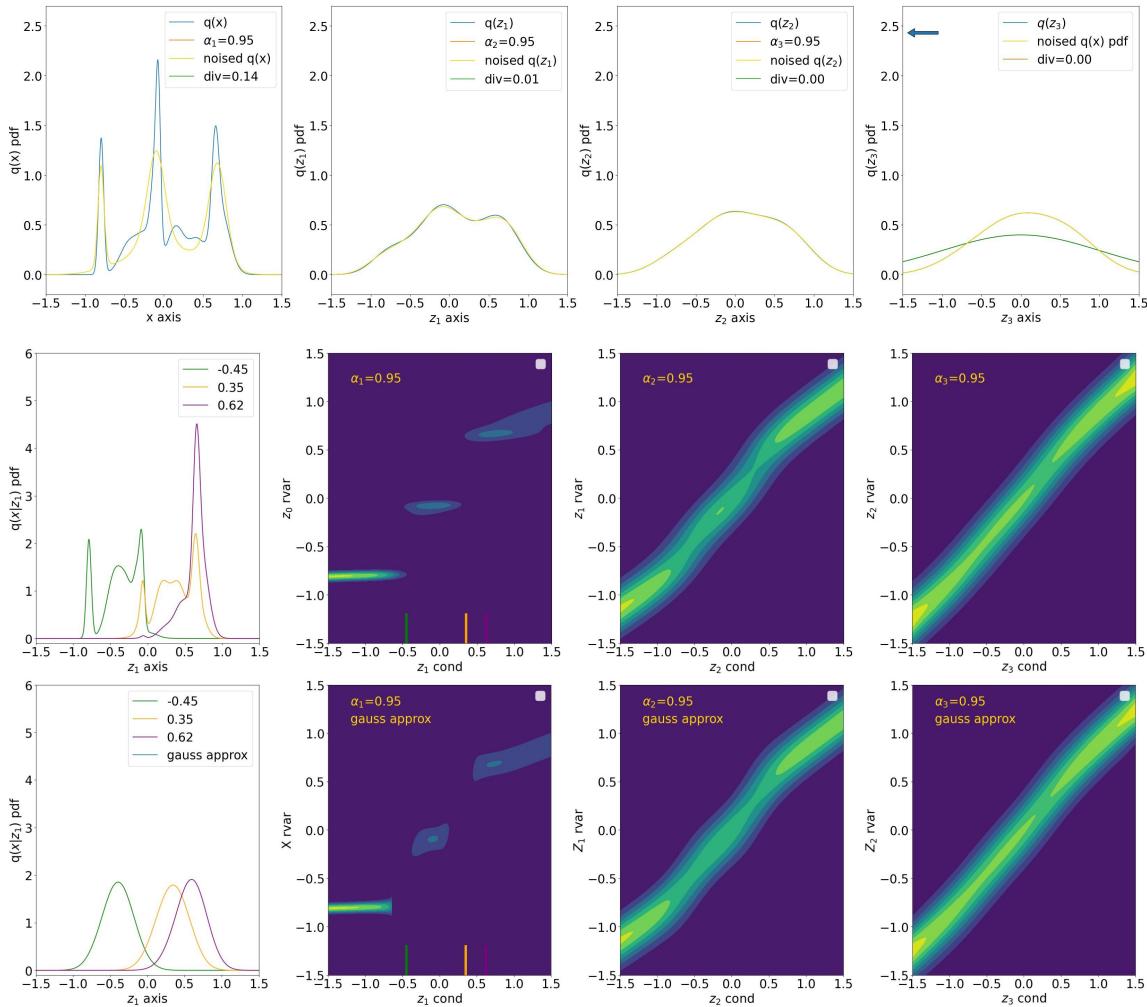


图 25: Gauss Fitting with $\alpha=0.95$

3.4.7 压缩映射 $q(z_{t-1}|z_t)$ 的不动点

3.4.5 节提到， $q(z_{t-1}|z_t)$ 是一个压缩映射 (Contraction Mapping)。根据 Banach Fixed Point Theorem，一个连续完备的压缩映射存在一个不动点。也就是说，对空间的任意点，只要连续作用足够多的 $q(z_{t-1}|z_t)$ 压缩映射，输出将稳定在某个固定点上，见示意图 27。经过多次实验发现，此变换的不动点恰好为 $q(z_{t-1})$ 。

图 30 是一个例子。第 1 子图和第 2 子图分别展示了 $q(x)$ 和 $q(z_1)$ 。 $q(x)$ 是一个随机初始的概率分布，经过加噪正变换后得到 $q(z_1)$ 。第 3 子图展示的是一个新的随机概率分布 (绿色曲线)，可以看出，与 $q(z_1)$ 差别较

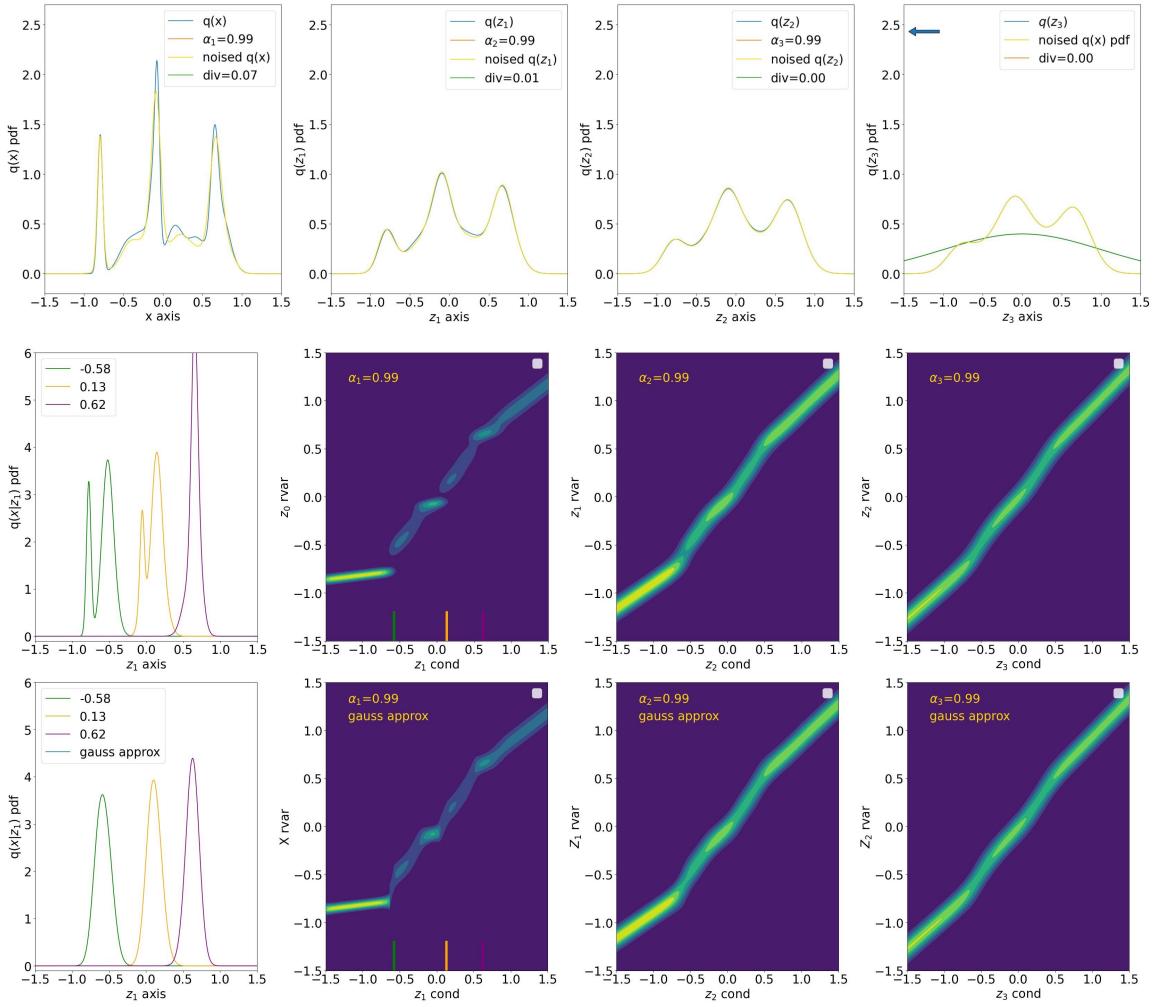


图 26: Gauss Fitting with $\alpha=0.99$



图 27: One Contraction Mapping Converges To Fixed Point

大。使用 $q(x|z_1)$ 逆变换对新的分布连续迭代作用，可以看出，起初输出的分布与 $q(x)$ 相似性较差，但随着次数增多，相似度越来越高，并在 60 次左右的时候趋向稳定。

图 31 是另外一个例子。此例子的 $q(x)$ 相对较复杂，起始的输入分布与 $q(x)$ 差别更大（第 3 子图），但经过 2000 次左右的迭代后，输出的分布也与 $q(x)$ 很接近。 $q(x)$ 越复杂，所需迭代的次数越多。

可能是由于数值计算近似误差的原因，虽然最终的分布与 $q(x)$ 挺相似，但还是略微有一些差别。差别的大小与 $q(x)$ 本身的复杂度和 α_t 的值（即噪声大小）有关。更进一步的问题待后面继续研究。

根据 Contraction Mapping 的性质，多个 Contraction Mapping 串联也是一个 Contraction Mapping。于是，可以把多个逆变换结合起来，作为一个大变换进行迭代，见示意图 28。这么做有什么好处呢？上面提到，一次“大的变换”可由多次“小的变换”等效替代，而“小的变换”更容易由 $p(z_{t-1}|z_t)$ 拟合，因此，循环迭代一串“小变换”能更好地恢复 $q(x)$ 。

图 32 是一个例子，使用 $\alpha = 0.95$ 的变换进行迭代，最终的拟合误差为 0.05(JS Div)。图 33 是另外一个例子，使用 10 个 $\alpha_t = 0.995$ 的变换作为迭代对象迭代，最终拟合误差也为 0.05(JS Div)。

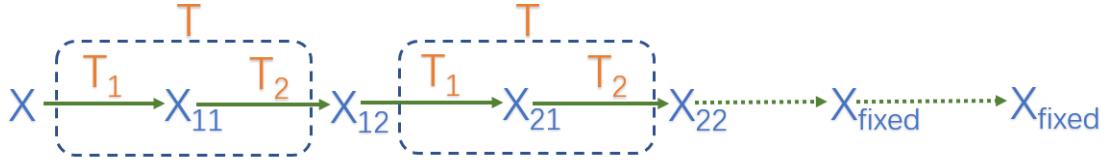


图 28: Sequence of Contraction Mapping Converges To Fixed Point

3.4.8 DPM 模型设计要点

基于上述的分析，为了让 $p(x, z_{1:T})$ 能更好地拟合 $q(x, z_{1:T})$ ，在 DPM 模型设计过程中，可参考几点建议。

- 不同的归一化方式应使用不同的参数策略，最好是将有效域归一化成中心对称的形式，与标准高斯分布的有效域较接近，比如 $[-1,1]$ 。
- 起始阶段，概率密度较复杂，应使用较大的 α_t (小噪声)；结尾阶段，概率密度较简单，可使用较小的 α_t (大噪声)，减少隐变量个数。
- 可考虑使用多个的均值映射函数 $\mu_{\theta_t}(z_{t+1})$ ，起始阶段，使用较复杂的；结尾阶段，使用较轻量的，可减少一定的计算量。
- 起始阶段，概率密度较复杂，可考虑使用可学习的方差，结尾阶段，概率密度较简单，可使用固定的方差。
- 最后一个隐向量的概率分布 $q(z_T|x)$ 应尽量接近于 $p(x_T)$ 。

3.4.9 DPM 模型的独特之处

由前面的分析可知，DPM 训练的过程是让各项 $p(z_{t-1}|z_t)$ 去拟合对应的 $q(z_{t-1}|z_t)$ ，此过程的思路与当前深度学习的方法有些差别，有自己独特的优点。

DPM 将一个大任务自动切分成多个独立训练的子任务。大任务是指把噪声分布转变成数据分布的复杂变换，子任务是指单独拟合一个 $q(z_{t-1}|z_t)$ 。由于 q 是一个固定概率分布，所以 $q(z_{t-1}|z_t)$ 是一个固定的函数，因此每项训练任务都有自己明确固定的目标，训练过程中不用考虑其它项的训练结果。这一点与神经网络有较大差别，神经网络各层之间存在耦合，难以单独训练。当层数较多时，神经网络存在梯度消失或爆炸的问题，而 DPM 模型不存在此问题。所以 DPM 模型可以包含 1000+ 个变换，每个变换都是近百层的神经网络。

3.4.10 是否可通过“逆卷积”恢复数据分布 $q(x)$

前面提到， $q(x)$ 到 $q(z_T)$ 的变换是由于一系列“收缩提升-卷积”变换组成。由于“收缩提升”和“卷积”均存在逆变换，所以理论上是可通过对应的逆变换从 $q(z_T)$ 恢复 $q(x)$ 。

但是，实际实现过程中会碰到一些问题。

噪声极度放大的问题

“逆卷积”变换具有极高的输入灵敏度。“逆卷积”变换的计算方式如下：

$$\begin{aligned} F(\omega) &= \mathcal{F}(f(x)) & G(\omega) &= \mathcal{F}(g(x)) & H(\omega) &= \mathcal{F}(h(x)) \\ f(x) \circledast g(x) &= h(x) & \Rightarrow & & F(\omega) * G(\omega) &= H(\omega) \\ f(x) &= \mathcal{F}^{-1} \left(\frac{H(\omega)}{G(\omega)} \right) \end{aligned} \tag{128}$$

其中 \mathcal{F} 代表傅里叶变换算子。

上述恢复信号 $f(x)$ 的过程，需要在频域除以卷积核的傅里叶变换 $G(\omega)$ 。此时，如果 $G(\omega)$ 存在一些接近 0 的值，计算将极不稳定，容易把细小的噪声无限放大，导致恢复失败。而在 q 模型的逆变换过程中，存在一个重要的近似，将 $q(z_T|x)$ 和 $q(z_T)$ 近似为 $\mathcal{N}(0, I)$ ，这便是明显的噪声，导致逆变换的起始阶段便引入大量的噪声。如果连续使用多个变换，噪声将被无限放大。

另外，也可以从另一个角度理解“逆卷积”的不可行性。由于 $q(z_{1:T}|x)$ 模型是确定的，所以卷积核（正变换）是固定的，“逆卷积”变换因此也是固定的。由于 $q(x)$ 可以是任意的分布，所以，通过一系列固定的“卷积”正变换，可以将任意的 $q(x)$ 转换成接近 $\mathcal{N}(0, I)$ 的分布。如“逆卷积”变换可行，则意味着，可用一个固定的“逆卷积”变换，将 $\mathcal{N}(0, I)$ 分布转换成任意的 $q(x)$ ，这明显是一个悖论。

根据经验和直觉总结，逆变换要有较好的抗噪性，需依赖于具体的输入和输出。

计算量的问题

对于低维的任务，傅里叶变换及反变换尚能实现。但在机器学习应用领域，维度一般都比较高（对于图像，经常大于 3000 维），傅里叶变换及反变换将难以实现。

采样困难的问题

通过“逆卷积”方式转换，只能得到概率密度分布 $q(x)$ ，不能以“祖先采样”的方法采样得到新样本。

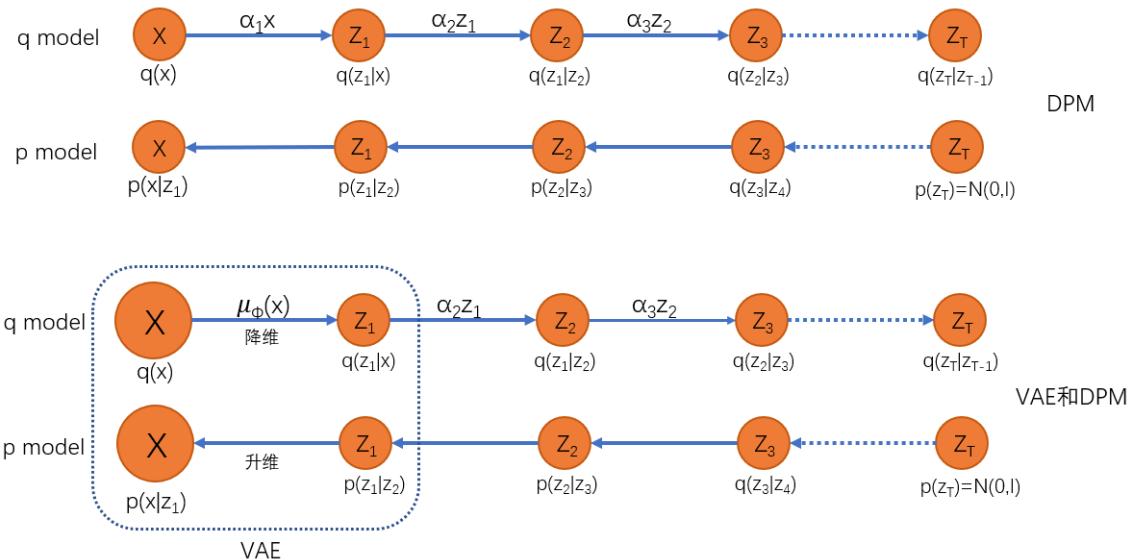


图 29: VAE And DPM

3.5 融合 DPM 模型和 VAE 模型

在当前的图像生成领域，Stable Diffusion 非常流行，其是一个两阶段模型。第一阶段，通过 VAE 对图像进行降维；第二阶段，对降维后的向量学习一个 DPM 模型。其实也可以把两个模型融合起来，进行联合训练或者微调 LORA。

对 DPM 模型作简单的改造可实现此目的。改造要点如下：

- 将 Z_1, Z_2, \dots, Z_T 隐变量的维度缩小，并保持相同。比如， $\dim(X) = 512 * 512$, $\dim(Z_i) = 64 * 64$ 。
- 将 $q(z_1|x)$ 的概率分布的均值修改为可学习的形式，并且是降维的。 $q(z_1|x) = \mathcal{N}(\hat{\mu}_\phi(x), I)$ 。 $\hat{\mu}_\phi(x)$ 可以是任意的函数，CNN 或者 Transformer 均可。
- 将 $p(x|z_1)$ 的概率分布的均值函数 $\mu_{\theta_1}(z_1)$ 修改为升维的形式。

改造前后的结构可参考图 29。

改造后， $q(z_t|x)$ 的概率分布形式略微有变化，需将 α_1 替换为 $\hat{\mu}_\phi(x)$ 。

$$q(z_t|x) = \mathcal{N}(\sqrt{\bar{\alpha}_t}x, 1 - \bar{\alpha}_t) \quad \text{where } \bar{\alpha}_t \triangleq \prod_{j=1}^t \alpha_j \quad \alpha_1 = \hat{\mu}_\phi(x) \quad (129)$$

把优化的目标函数 $\mathcal{L}(\theta, \phi)$ 重新写出来

$$\mathcal{L}(\theta, \phi) = \underbrace{\int q(z_1|x) \log p(x|z_1) dz_1}_{L_1} - \sum_{t=2}^T \underbrace{\int q(z_t|x) \overbrace{KL[q(z_{t-1}|z_t, x) \| p(z_{t-1}|z_t)]}^{KL_t} dz_t}_{L_t} - \underbrace{KL[q(z_T|x) \| p(z_T)]}_{L_{T+1}} \quad (130)$$

分析各项的特点：

- 起始项 L_1 项与 VAE 的重建项相同， $q(z_t|x)$ 是可学习的。
- 结束项 L_{T+1} 由于依赖于 $\bar{\alpha}_t$ ，所以其不能当作常数项对待。但如果隐变量的个数特别多，由于 $\bar{\alpha}_t$ 的累乘效应，最后隐变量的概率分布 $q(z_T|x)$ 也趋向于标准高斯分布，故此式的值也会比较小，也可忽略不考虑。
- 中间项还是维持 DPM 模型形式，但 $q(z_t|z_t, x)$ 的均值依赖于 $\bar{\alpha}_t$ ，所以其不再是常数，因此，式(92)和式(99)中关于 $\mu_{\theta_t}(z_t)$ 的重参方式不再可用。所以，融合后的模型不能以 **Predict X** 和 **Predict Noise** 的方式进行优化，只能按 **Predict Next Step** 的方式进行优化。

参考文献

- [1] Auto-Encoding Variational Bayes
- [2] Deep Unsupervised Learning Using Nonequilibrium Thermodynamics
- [3] Denoising Diffusion Probabilistic Models
- [4] Products and Convolutions of Gaussian Probability Density Functions
- [5] Complete Probabilistic Metric Spaces
- [6] Transformations of Random Variables - <https://stats.libretexts.org/>
- [7] Pattern Recognition and Machine Learning.

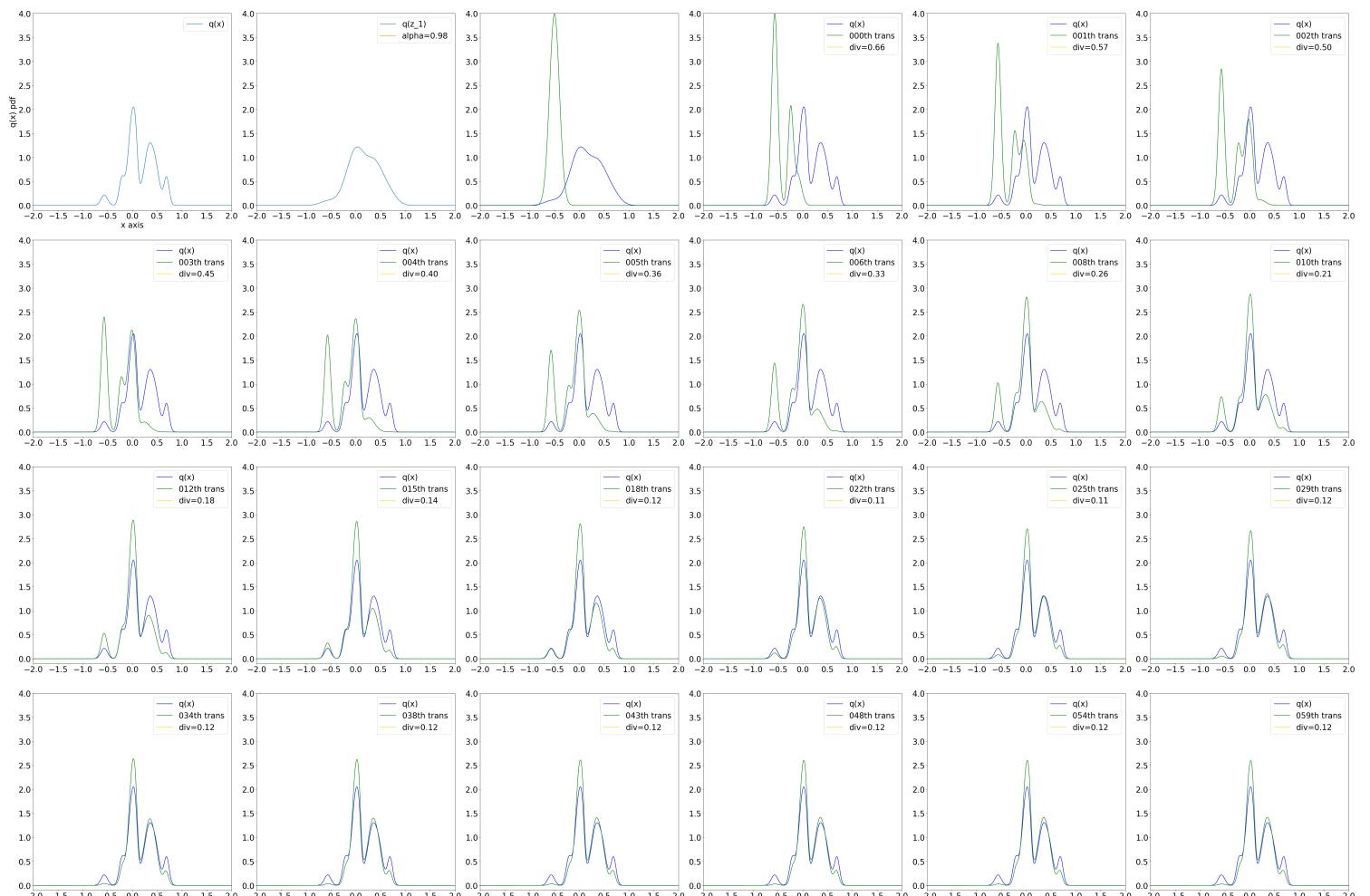


图 30: Converging to Fixed Point(60 iteration)

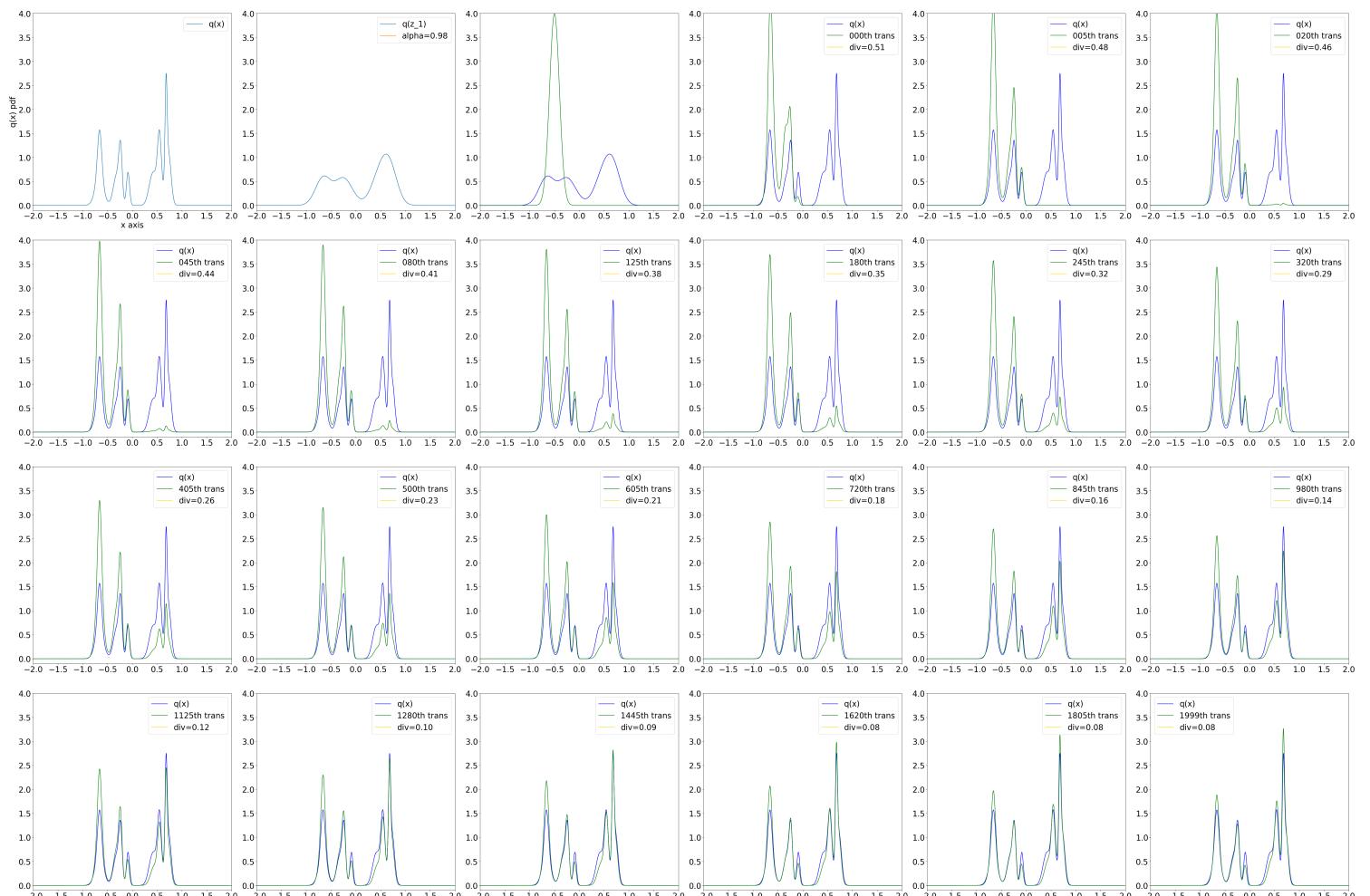


图 31: Converging to Fixed Point(2000 iteration)



图 32: Iterate trans

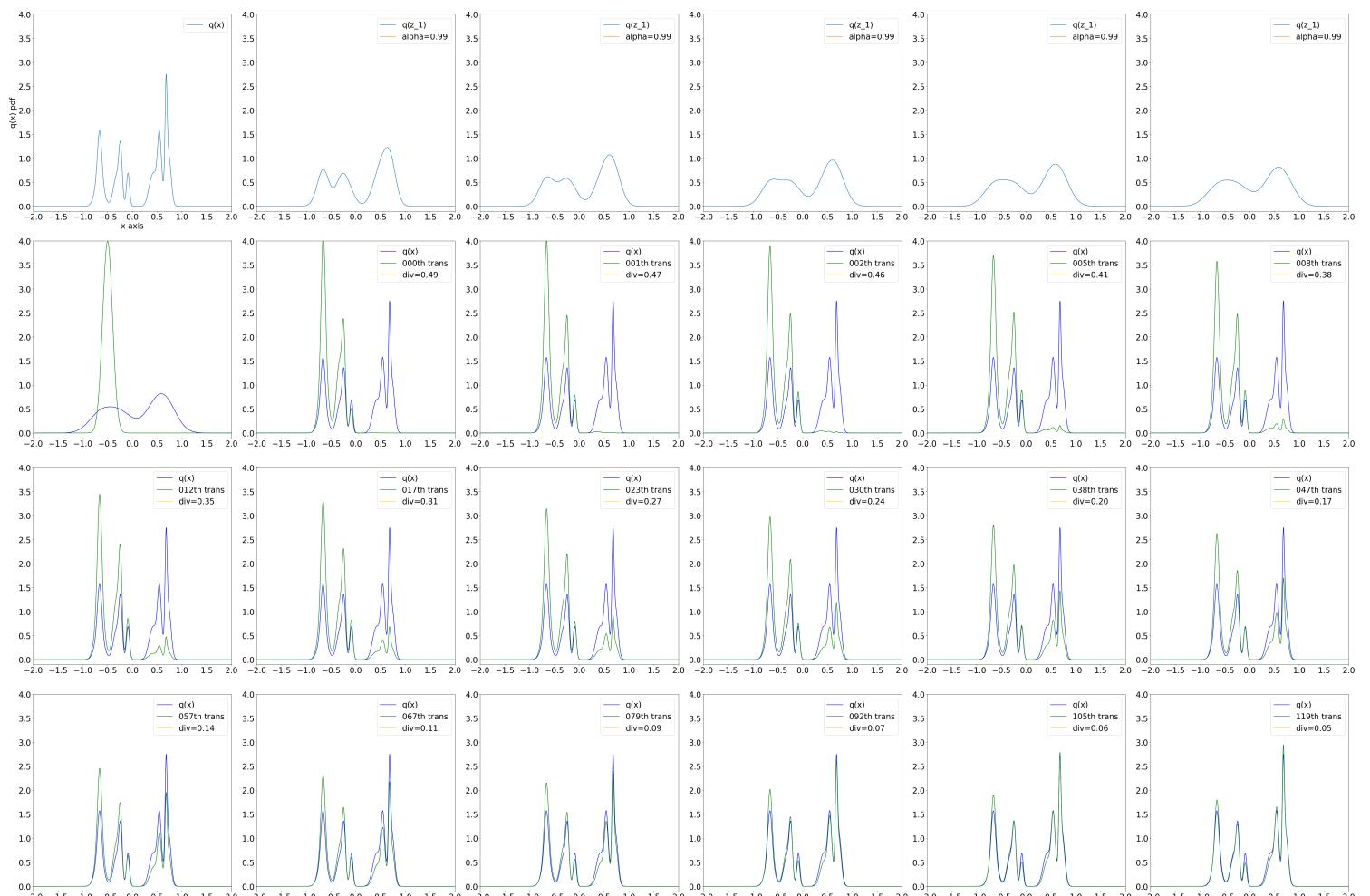


图 33: Iterate sequence of trans