VAE 简明教程

日常半躺

2022 年 10 月 8 日

本文希望做一个深入浅出的 VAE 教程，既通俗易懂又深入讲解数学原理。

1 基本概念

先验和后验。 先验是说一个随机变量已知的分布，通常在机器学习里代表人工给定的条件，比如 $p(z) \sim \mathcal{N}(0, I)$。后验是说在给定一些条件（或者叫知识）以后先验发生了偏移，偏移后的概率称作后验，因此后验是相对于先验来说的，比如 $p(z|x) \sim \mathcal{N}(f(x), g(x)I)$。

2 ELBO（Evidence Lower-bound）

给定一个数据集 $\{x_1, x_2, \ldots, x_n\}$，在机器学习中会用到两个常用的概念，一个是假设每个样本点是**独立同分布**的，第二个是在学习模型的时候使用**最大似然估计** $\underset{\theta}{\operatorname{argmax}} \prod p(x_i)$。实际跑模型的时候，优化的目标是最大化对数似然 $\underset{\theta}{\operatorname{argmax}} \sum \log p(x_i)$。但是，直接优化这个目标在很多时候不可行，于是在 VAE 系列模型中我们转而最大化对数似然的一个下界，这个下界被称为 ELBO。

$$
\begin{aligned}
\log p(x) &= \log \int_{\mathcal{Z}} p(x|z)p(z)dz && (1)\\
&= \log \int_{\mathcal{Z}} q(z)\frac{p(x|z)p(z)}{q(z)}dz && (2)\\
&= \log E_{z\sim q(z)}[\frac{p(x|z)p(z)}{q(z)}] && (3)\\
&\geq E_{z\sim q(z)}[\log \frac{p(x|z)p(z)}{q(z)}] && \text{(Jensen inequity)} \quad (4)\\
&= E_{z\sim q(z)}[\log p(x|z)] - KL(q(z)\|p(z)) && (5)
\end{aligned}
$$

也就是说，对于给定的样本 x_i，我们希望寻找一个分布 $q(z)$ 和一个分布 $p(x|z)$，使得代入 $x = x_i$ 时的 ELBO 越大越好。需要注意的是，虽然这里的 $q(z)$ 是给定 x 以后找到的，但是为了简便一般不写成 $q(z|x)$。

3 VAE

有了优化目标，我们来看一下如何寻找这两个分布 $q(z)$ 和 $p(x|z)$。原始的 VAE 加入了以下三个假设：

1. $q(z) \sim \mathcal{N}(Enc_\mu(x;\theta), Enc_{\sigma^2}(x;\theta)I)$
2. $p(z) \sim \mathcal{N}(0, I)$
3. $p(x|z) \sim \mathcal{N}(Dec(z;\phi), I)$

通俗地说，由于我们也不知道如何寻找这两个分布才能让 ELBO 最大，于是我们就用两个神经网络去预测 $q(z)$ 和 $p(x|z)$。其中，给定 x 以后，encoder 神经网络 Enc_μ 和 Enc_{σ^2} 可以预测出最优的 $q(z)$，decoder 神经网络 Dec 可以根据拿到的分布 $q(z)$ 计算最优的 $p(x|z)$。

这里有两个非常关键的点，绝大多数教程都没有明确指出来。

- 第一点：**如何让神经网络输出概率分布**。我们知道，神经网络的输出是若干个确定的数值，那怎么用数值去表示分布呢？必须引入人工给定的假设，这就是上面的假设 1 和假设 3 产生的原因。由于这两个假设的存在，我们把最优的分布限定在了正态分布里。因此只需要用神经网络输出对应的均值和方差就可以了。

- 第二点：**如何给神经网络输入概率分布**。上一段我们说"decoder 神经网络 Dec 可以根据拿到的分布 $q(z)$ 计算最优的 $p(x|z)$"，也就是说 decoder 是以概率分布作为输入的，但是我们知道神经网络的输入也是确定的数值，所以怎么办呢？答案是采样。直接面向目的入手，我们为什么想要得到 $p(x|z)$ 呢？其实真正的目的是估计 $E_{z\sim q(z)}[\log p(x|z)]$（也就是 ELBO 的第一项）。现在我们已经有了 $q(z)$，就可以用采样的方法直接估计这个期望，我们的神经网络只需要负责把一个采样的 z 转换为一个关于 x 的概率分布就可以了。

至此，神经网络的前向传播过程就打通了，如下图所示：

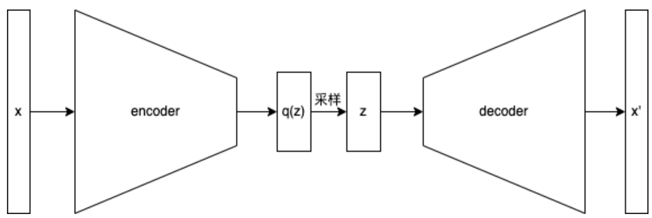

图 1: VAE 前向传播

接下来我们来看一下基于 VAE 的三个假设，得到的 ELBO 具体的表达式是什么。

先回顾以下 ELBO 的公式：

$$\log p(x) \geq E_{z\sim q(z)}[\log p(x|z)] - KL(q(z)\|p(z)) \tag{6}$$

再回顾一下 VAE 的三个假设：

1. $q(z) \sim \mathcal{N}(Enc_{\mu}(x;\theta), Enc_{\sigma^2}(x;\theta)I)$
2. $p(z) \sim \mathcal{N}(0, I)$
3. $p(x|z) \sim \mathcal{N}(Dec(z;\phi), I)$

根据假设 1 和假设 2，可以得到 ELBO 的第二项。设 $\mu = Enc_{\mu}(x;\theta)$、$\sigma^2 = Enc_{\sigma^2}(x;\theta)$，有

$$KL(q(z)\|p(z)) = \frac{-\log\sigma^2 + \sigma^2 + \mu^2 - 1}{2} \tag{7}$$

这是因为两个正态分布的 KL 散度为：

$$KL(p\|q) = \log \frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{2\sigma_2^2} - \frac{1}{2} \tag{8}$$

根据假设 3，可以得到 ELBO 的第一项：

$$E_{z\sim q(z)}[\log p(x|z)] = E_{z\sim q(z)}[\log \frac{1}{\sqrt{2\pi}} \exp(-\frac{\|x - Dec(z;\phi)\|^2}{2})] \tag{9}$$

$$= E_{z\sim q(z)}[-\frac{\|x - Dec(z;\phi)\|^2}{2}] + C \tag{10}$$

综上，最大化 ELBO 等价于最大化如下表达式：

$$E_{z\sim q(z)}[-\|x - Dec(z;\phi)\|^2] + \log \sigma^2 - \sigma^2 - \mu^2 \tag{11}$$

现在只剩最后一个问题，反向传播优化。根据图 1的前向传播过程，采样操作是不可导的，因此反传到这里梯度会失效。于是这里引入了一个技巧——重参数。从一个均值为 μ 方差为 σ^2 的正态分布里采样 z，可以转化为先从 $\mathcal{N}(0,I)$ 采样 t，然后令 $z = \mu + t\sigma$，这样采样的过程就可导了。

4 输出 0/1 的 VAE

理解原理的好处就是对各种变体也可以自己推导。比如这里我们假设 x 不再是实数，而是离散的只有两种可能的取值 0/1，比如黑白图片。我们只需要修改上面的假设 3，将正态分布改成伯努利分布：

3. $p(x|z) \sim B(Dec(z;\phi))$

我们让 decoder 输出的值代表伯努利取 1 的概率，那么当 $x = 1$ 时，$\log p(x|z) = \log Dec(z;\phi)$；当 $x = 0$ 时，$\log p(x|z) = \log(1 - Dec(z;\phi))$。合并以后就得到了 ELBO 的第一项表达式：

$$\log p(x|z) = x \log Dec(z;\phi) + (1-x)\log(1 - Dec(z;\phi)) \tag{12}$$

恰好就是将最小均方误差损失（MSE）变成了二元交叉熵损失（BCE）。

有趣的是，这种 BCE 的损失对于输入 x 在 0 到 1 区间实数的情况也适用，比如灰度图片。

$$\log p(x|z) = x \log Dec(z;\phi) + (1-x)\log(1-Dec(z;\phi)) \tag{13}$$
$$= \log Dec(z;\phi)^x (1-Dec(z;\phi))^{1-x} \tag{14}$$

于是这个操作相当于把假设 3 改为：

3. $p(x|z) \sim G(Dec(z;\phi))$，其中分布 G 的概率密度函数是 $p(x|z) = \frac{Dec(z;\phi)^x(1-Dec(z;\phi))^{1-x}}{\mathcal{Z}}$，$\mathcal{Z}$ 是归一化常数。

直观感受一下这个概率密度函数的样子：

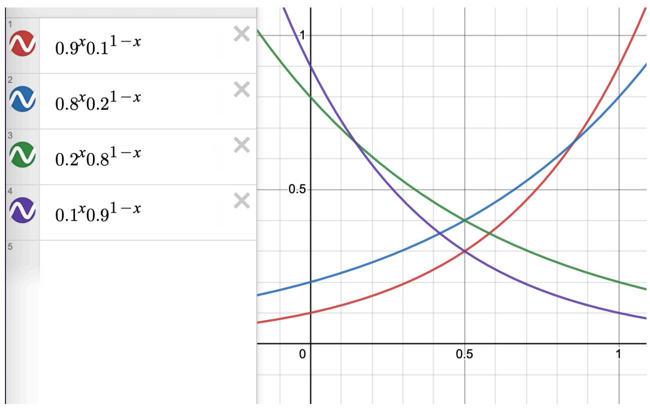

图 2: G 的概率密度函数可视化

因此，如果用 BCE 损失函数训练 0 到 1 区间的 VAE，当 decoder 给出了 0.8 这个值，并不代表 0.8 概率是最大的，而是 1 的概率是最大的，因此贪心策略的话应该输出 1。即最优的策略是大于 0.5 输出 1、小于 0.5 输出 0。

再拓展一下，如果输出的是经过 softmax 的概率分布，而不是 sigmoid 激活的单个概率，又该怎么做呢？假如有 3 个类别，真实标签是 one-hot 的，decoder 预测的输出是一个经过 softmax 的概率分布。将假设 3 修改如下：

3. $p(x_0, x_1, x_2|z) \sim$ 依照 $Dec(z;\phi)$ 进行采样的概率分布 (p_0, p_1, p_2)。

也就是说，$(x_0 = 1, x_1 = 0, x_2 = 0)$ 的概率是 p_0，$(x_0 = 0, x_1 = 1, x_2 = 0)$ 的概率是 p_1，$(x_0 = 0, x_1 = 0, x_2 = 1)$ 的概率是 p_2。合并一下就是：

$$\log p(x_0, x_1, x_2|z) = x_0 \log p_0 + x_1 \log p_1 + x_2 \log p_2 \tag{15}$$

因此，实现的时候可以直接用类似 CrossEntropyLoss 和 NLLLoss 来优化。

5 AE

如果我们修改一下假设 1：

1. $q(z) \sim \mathcal{N}(Enc(x;\theta), I)$

根据公式 11，修改假设 1 以后最大化 ELBO 等价于最大化如下表达式：

$$E_{z\sim q(z)}[-\|x - Dec(z;\phi)\|^2] - [Enc(x;\theta)]^2 \tag{16}$$

也就是相当于最小化

$$E_{z\sim q(z)}[\|x - Dec(z;\phi)\|^2] + [Enc(x;\theta)]^2 \tag{17}$$

发现什么了吗? 这其实就是 auto-encoder 的损失函数加上了一个 L2 正则项!

6 VQ-VAE

VQ-VAE 对 z 的空间做了更强的限制，认为 z 只能取 k 个向量中的某一个，这里的 k 就是词表的大小。在此基础上做了如下三个假设：

1. $q(z) \sim$ 用 $Enc(x;\theta)$ 得到的向量与词表中的 k 个向量 L2 距离最小的那个向量概率为 1，其余概率为 0
2. $p(z) \sim$ k 个向量概率均匀分布
3. $p(x|z) \sim \mathcal{N}(Dec(z;\phi), I)$

再回顾一下 ELBO 的公式：

$$\log p(x) \geq E_{z\sim q(z)}[\log p(x|z)] - KL(q(z)\|p(z)) \tag{18}$$

单点分布和 k 个点的均匀分布的 KL 散度是 $\log k$，也就是一个常数。因此最大化 ELBO 等价于最大化第一项期望，也就是

$$E_{z\sim q(z)}[\log p(x|z)] = E_{z\sim q(z)}[\log \frac{1}{\sqrt{2\pi}} \exp(-\frac{\|x-D(V[z];\phi)\|^2}{2})] \tag{19}$$

$$= -\frac{\|x-D(V[z];\phi)\|^2}{2} + C \tag{20}$$

其中 $V[z]$ 表示经过 encoder 编码以后在词表中选中的向量。

因此，最大化 ELBO 等价于最小化输入和输出的最小均方误差。中间的采样过程同样会导致梯度断开，这里没有使用重参数，而是直接把选中的向量的梯度 copy 给 encoder 的输出，同时还在 loss 里加入了词表和 encoder 输出的 MSE（事实上这里比重参数还要 trick），具体可参照代码。

7 最后

最后还剩一个问题。上面我们讨论的结果是用 encoder 和 decoder 拟合两个分布 $q(z)$ 和 $p(x|z)$ 使得 ELBO 最大，但是为什么 decoder 可以用来生成图像呢？答案是没有任何数学上的保证让 decoder 可以生成真实图片，但是我们可以根据优化目标（ELBO）来直观地解释一下。

$$\log p(x) \geq \underbrace{E_{z\sim q(z)}[\log p(x|z)]}_{reconstruction} - \underbrace{KL(q(z)\|p(z))}_{regularization} \tag{21}$$

经过优化以后，第一项 reconstruction 会尽可能大，也就是给定 z 以后 decoder 的输出会尽可能让真实样本的概率大；第二项 regularization 会尽可能小，也就是给定 x 以后 encoder 的输出会尽可能接近先验分布。

综合上面的两项，得到的结果就是**从近似先验的分布里去采样** z 传给 decoder，可以**产生近似真实样本**的输出，也就是图片生成效果。