# Variational Autoencoder 学习笔记

秦震岳

2018 年 11 月 21 日

Variational autoencoder, 简称 VAE, 是一种生成式模型. 顾名思义, 生成式模型是为了去生成某种事物. 比如, 去生成数字 0 到 9 的手写图片.

VAE 中的一个关键概念为 latent variable 和 latent space. 这两个概念高度相关, 具体来讲, latent variable 是 latent space 中的变量. 这些 latent variable 可以被看做是事物 $X$ 的决定特征. 这些 latent variable 的维度一般比其所对应的事物 $X$ 的维度要小, 因此从 $X$ 到 $z$ 有一定的压缩在里面. 换句话说, 他们决定了 $X$ 之所以被叫做 $X$ 的决定属性. 比如, 如果 $X$ 是一个人的相片, 那么我们可以确信这个相片有一个头, 一个身体等等. 当然, 每个人的头长得也不太一样, 所以 latent variable 也不尽相同, 用来表示事物 $X$ 之间的变化. 我们用 $z$ 来表示 latent variables, $Z$ 来表示 latent space.

我们的目标就是找到 $X$ 和 $z$ 之间的概率关系. 显然, 如果 $z'$ 是 $X'$ 所对应的 latent variable, 我们假设这样一个概率 P 受参数 $\theta$ 所控制, 我们希望概率 $P(X'|z', \theta)$ 可以得到一个较高的数值. 总体来说, 我们希望对于每一个 $X$ 来说,

$$P(X) = \int P(X|z;\theta)P(z)dz \tag{1}$$

都能够取到最大值. 为了整洁, 接下来我们在书写时, 会省略参数 $\theta$.

现在我们的目标有了, 但是想要确定 $P(X)$ 并非易事. 有两个问题: 第一, 如何确定 $P(X|z)$ 和 $p(z)$; 第二, 如何做积分计算.

我们先来讨论确定 $P(X|z)$ 和 $p(z)$ 的问题. 人为地确定 $z$ 中每一个维度是什么意思是不现实的, 因为这首先需要非常大量的人力, 并且生成的结果质量也未必能令人满意. 幸运地是, 之前的研究从理论上证明了神经网络输出的结果套上一个高斯分布可以模拟任何的分布. 因此, 我们使用神经网络加上高斯分布来模拟 $P(X|z)$, 形式化可以表示为[1]

$$P(X|z) = \mathcal{N}(X|f(z), \sigma^2 \cdot I)$$

一般地, 如果我们可以成功地从 $z$ 还原出 $X$, 我们就可以说 latent strucure 存在[2]. 同时, $P(z)$ 可以被表示为 $\mathcal{N}(z|0, I)$[3].

接下来我们来考虑怎么样求积分的问题. 因为 $z$ 是用来记录所对应的 $X$ 的特征, 所以对于某个 $X$, 大部分 $z$ 应该是用不上的. 换句话说, 对于大多数的 $z$, $P(X|z)$ 几乎都是 0. 我们想利用这一点来减小需要考虑的 $z$ 的空间. 为此, 我们引入一个新的函数 $Q(z|X)$. 这个函数对于一个输入 $X$, 返回比较有可能和其有关的 $z$. 换句话说, $Q(z'|X')$ 中如果 $z'$ 和 $X'$ 有关系, $Q(Q(z'|X'))$ 将会返回一个比较大的数值. 因此, 如果 Q 足够好的话, 我们应该有 $E_{Z\sim Q}P(X|z) = P(X)$. 实际上, $E_{Z\sim Q}P(X|z)$ 和 $P(X)$ 的关系是整个 variational autoencoder 的基石.

[1] ASK: Why the covariance in the Gaussian distribution can be diagonal?

[2] ASK: Any proof for this conclusion?

[3] ASK: Why?

我们现在来讨论 $E_{Z\sim Q}P(X|z)$ 和 $P(X)$ 的关系. 不难看出, $E_{Z\sim Q}P(X|z)$ 和 $P(X)$ 都是对于 $X$ 分布, 所以很自然的想法是去算一下它们两者的 KL divergence. 再一次强调, 我们想要 $Q(z|X)$ 和 $P(z|X)$ 分布大致相同, 所以我们首先计算它们两个之间的 KL-divergence, 然后再利用贝叶斯定理去计算 $P(X|z)$. 根据 KL-divergence 的定义, 我们有

$$\mathcal{D}[Q(z|X)||P(z|X)] = E_{Z\sim Q}[\log Q(z|X) - \log P(z|X)] \tag{2}$$

我们对 P(z|X) 使用贝叶斯定理, 可以得到

$$\mathcal{D}[Q(z|X)||P(z|X)] = E_{Z\sim Q}[\log Q(z|X) - \log P(X|z) - \log P(z)] + \log P(X) \tag{3}$$

通过移项和整理, 我们进一步有

$$\log P(X) - \mathcal{D}[Q(z|X)||P(z|X)] = E_{Z\sim Q}[\log P(X|z)] - \mathcal{D}[Q(z|X)||P(z)] \tag{4}$$

等式4是 variational autoencoder 的关键. 它可以用作目标函数, 并且等式左边是我们想要最大化的目标. 具体来说, 首先, 最大化 $\log P(X)$ 是生成式模型的目标, 请不要忘了我们的根本目的是使得我们的模型总体上最有可能生成我们所见过的每一个 $X$; 其次, $\mathcal{D}[Q(z|X)||P(z|X)]$ 是我们对于选择了不好的 $Q$ 的惩罚. 这个式子更加令人欣喜的另一点是如果 Q 已经给定, 我们可以用随机梯度下降 (SGD) 来优化等式右边[4]. 如果我们仔细观察的话, 等式右边本身长得就像一个 autoencoder: 第一项为从 $X$ 到 $z$ 的 encoding 和从 $z$ 到 $X$ 的 decoding 的质量; 第二项衡量当我们用 $Q$ 来代表 $P$ 来减小 latent space 时, 我们丢失了多少信息.

现在我们有了目标, 接下来的问题就是怎么样通过等式右边来进行优化. 外面那个 $E_{Z\sim Q}$ 现在处理起来有点困难, 所以我们暂时先无视它. 我们现在考虑在已经知道 $z$ 的基础上, 如何计算 $P(X|z)$ 和 $\mathcal{D}[Q(z|X)||P(z)]$. 在上文, 我们已经求出来了 $P(z)$. 我们再次利用 "神经网络的结果套上一个高斯分布可以模拟任何分布的特性", 将 $Q(z|X)$ 表示为 $\mathcal{N}(z|\mu(X), \Sigma(X))$. 在该式子中, $\mu(X)$ 和 $\Sigma(X)$ 由神经网络来确定. 到此, 我们已经能够计算出 $\mathcal{D}[Q(z|X)||P(z)]$. 在上文, 我们也已经算出来了 $P(X|z)$. 因此, 如果不考虑外面的 $E_{Z\sim Q}$, 我们已经能够确定 $[\log P(X|z)] - \mathcal{D}[Q(z|X)||P(z)]$.

现在我们来考虑 $E_{Z\sim Q}[\log P(X|z)] - \mathcal{D}[Q(z|X)||P(z)]$, 请注意如果我们对此式求梯度, 梯度会被移动到期望里面. 所以, 我们的策略是不断取样和随机梯度下降, 最终 $[\log P(X|z)] - \mathcal{D}[Q(z|X)||P(z)]$ 会收敛为 $E_{Z\sim Q}[\log P(X|z)] - \mathcal{D}[Q(z|X)||P(z)]$.

[4] ASK: When can we use SGD to do the optimization?