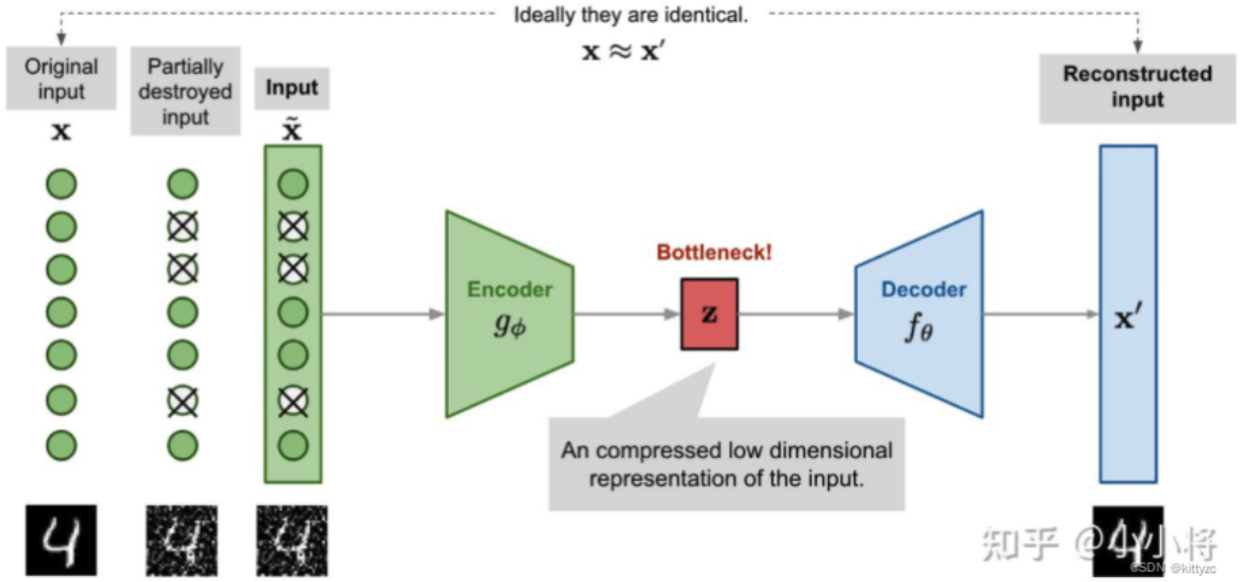
图像生成实践报告

— 饶翔云,520030910366

0. VAE原理

VAE（Variational AutoEncoder）是一种根据概率分布的生成式模型。要理解VAE模型的原理，首先要从最基本的AutoEncoder开始理解。



AutoEncoder的设计初衷是为了数据降维，假设原始特征x维度过高，那么我们希望通过编码器E将其编码成低维特征向量z=E(x)，编码的原则是尽可能保留原始信息，因此我们再训练一个解码器D，希望能通过z重构原始信息，即x≈D(E(x))。而他的目标优化函数可以被以下公式表示：

$$\min_{E,D} f(X,E,D) = ||X - D(E(X))||^2$$

即我们所谓的重建误差（Reconstruct Loss）。在VAE中，这种重建的思想被保留下来。

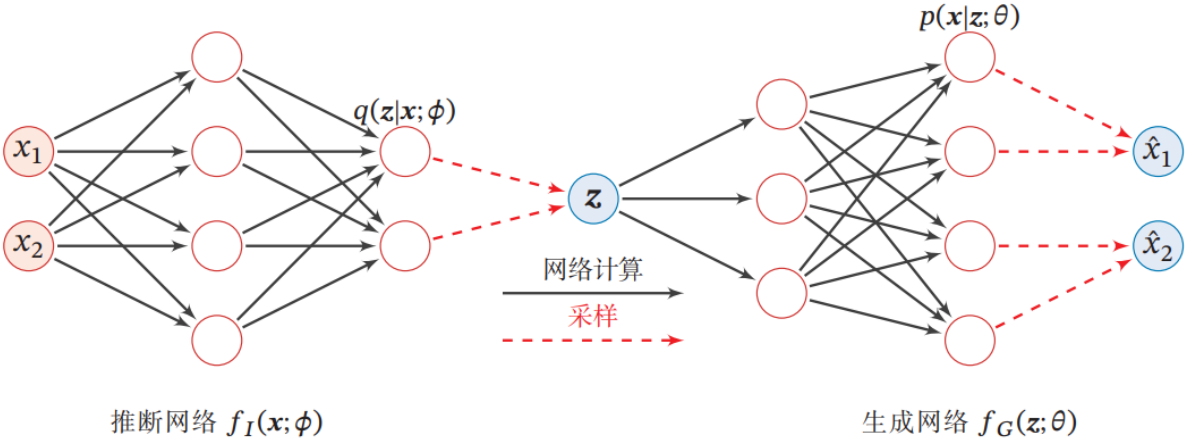
因为我们想要使得隐向量服从标准正态分布，即均值为0，标准差为1的正态分布，所以需要通过优化KL散度来使得分布逼近标准正态分布。其中KL散度的计算公式为：

$$KLD(p,q) = \log\frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_1^2 + \mu_2^2)}{2\sigma_2^2}$$

而VAE与AE不同的地方在于，VAE充分利用了中间的Bottleneck（记为z），并采取了随机性的思想。VAE的前提假设是输入变量X通过Encoder部分得到了一个隐变量z，这个z被认为是从标准正态分布中采样获得（随机性所在），decoder根据这个隐变量来重建信息。换而言之，z其实是一个条件变量，根据z的不同，环境条件不同，生成的X也不同。即：

$$P(X) = \sum_z P(X|z)P(z)$$

VAE颇具魅力的一点就在于随机性。但由于重构过程受到噪声的影响，因为Z是重新采样过的，不是直接由encoder算出来的。显然噪声会增加重构的难度，不过好在这个噪声强度(也就是方差)通过一个神经网络算出来的，所以最终模型为了重构得更好，肯定会想尽办法让方差为0。而方差为0的话，也就没有随机性了。但是VAE为了避免随机性丧失，它采用了让采样后的z服从标准正态分布的方法，即$P(z|X)\sim N(0,1)$。如果满足该条件，那么$P(z)=\sum_z P(X)P(z|X)\sim N(0,1)$。由此，VAE保证了模型的随机性不会消失，他会根据不同的输入生成不同的结果。

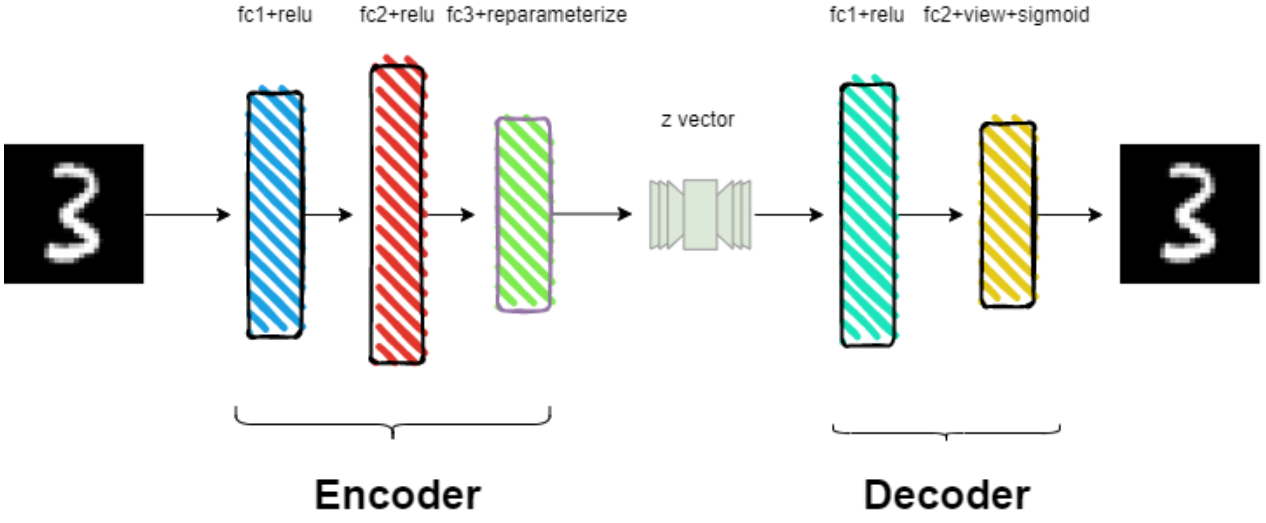


总而言之，VAE通过Encoder，生成μ和σ，然后通过采样得到z向量，让Decoder在由z向量所对应的概率空间生成重建后的结果。值得一提的是，VAE的优化目标是最小化重建误差和z服从的高斯分布（$N(\mu,\sigma^2)$）与标准正态分布之间的KL散度。为了实现KL散度可微，我采取了一点小trick。

```
1  z = mu + logvar * torch.rand_like(logvar)
```

1. 模型架构

我设计的模型结构如下：



模型参数如下：

	fc1	fc2	fc3
Encoder	28*28, 256	256,1024	1024,z_dim
Decoder	z_dim,28*28	28*28, 28*28	None

2. 实验过程

使用超参数：

lr	epochs	optimizer	batch-size
1e-3	100	Adam	32

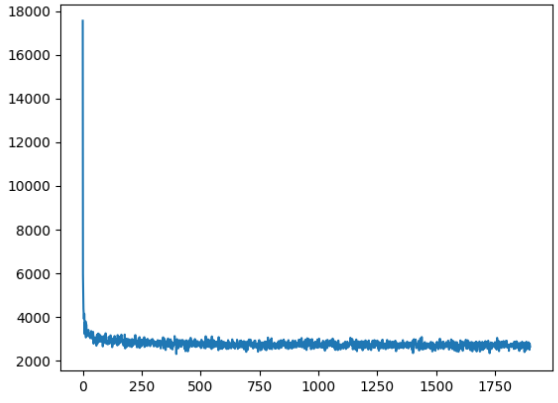
使用BCEloss和KLDloss混合的新loss进行训练。即：

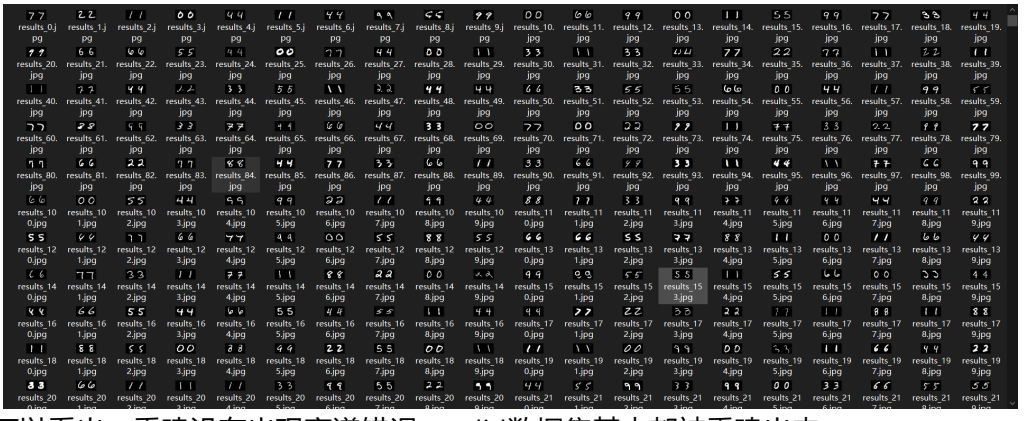
$$loss = BCEloss + KLDloss$$

$$KLDloss(\mu, \sigma) = -\frac{1 + \sigma - \mu^2 - \sigma^2}{2}$$

3. 实验结果(best)：

当隐层维度设为20的时候，重建获得了最佳效果：





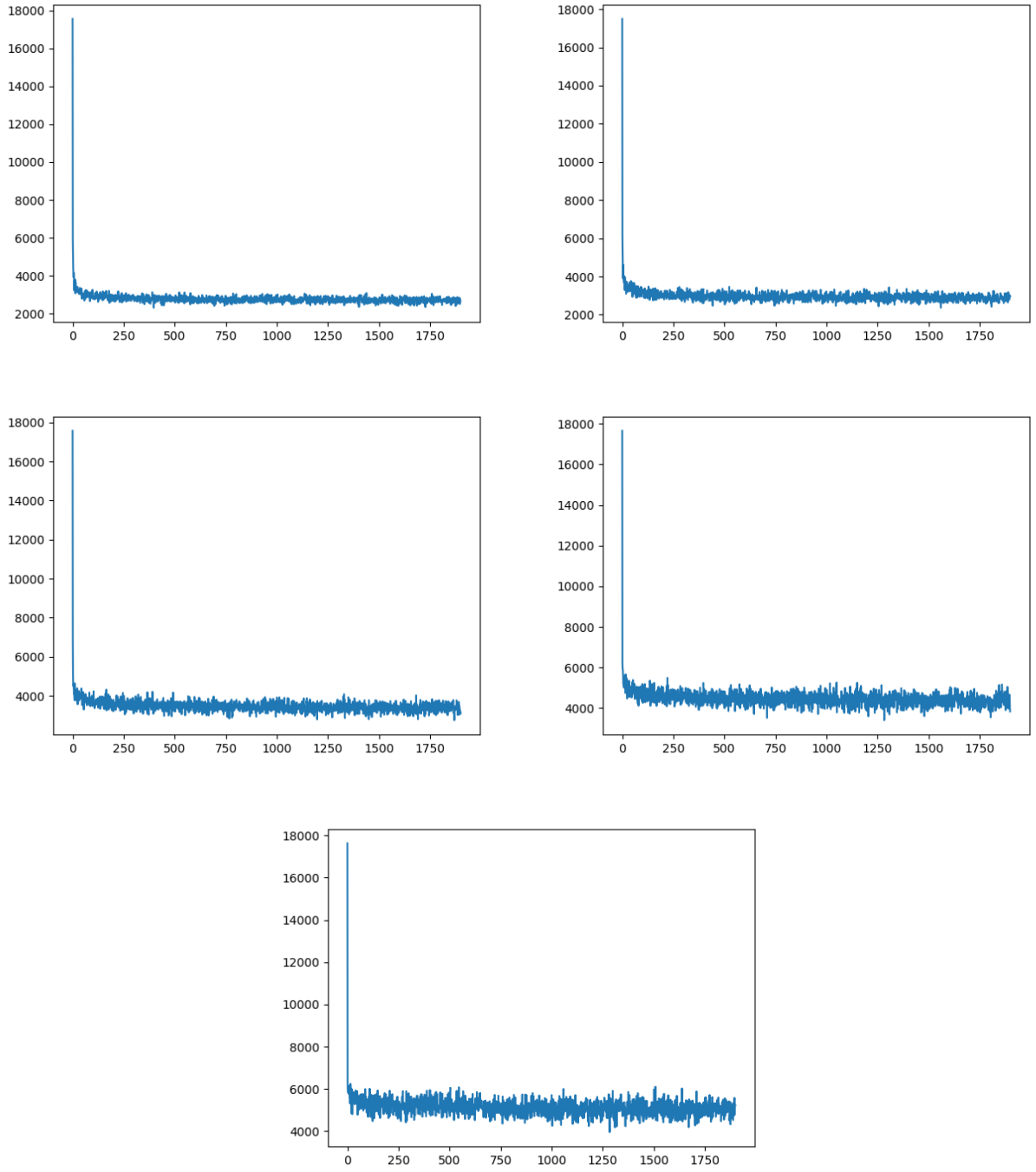
从图中可以看出，重建没有出现离谱错误，valid数据集基本都被重建出来。

4. 探究部分：

a. 对z_dim的探究。

我分别取z_dim为[1,2,5,10,20],进行模型训练并检查重建结果。

loss曲线分别如下：



可以从中看出，当z_dim高的时候，总loss可以得到一个较低的值，而z_dim低的时候，总loss高，而且波动大。而且根据我对重建结果的观察，当z_dim取高值的时候，重建结果非常的好，而当z_dim低的时候，得到的重建结果就会变得没有意义（一般只能还原出0,1等简单数字）。

b. 将隐层向量z维度设置为1，比较VAE训练完成后不同的z值对应的生成图片效果。

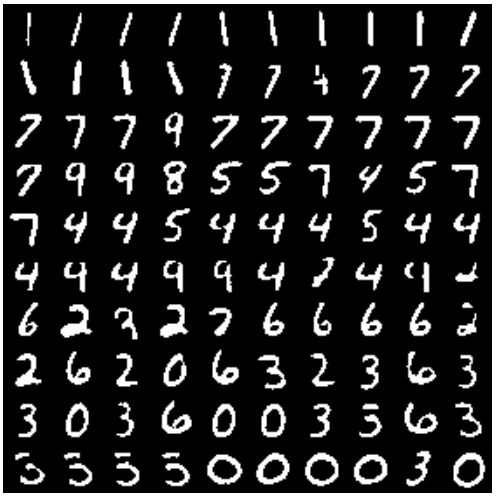
根据我的观察，我做出以下总结：

z值在0附近，Decoder会倾向于生成像4的图形；在0.1附近，Decoder会倾向于生成像9（7）的图形；在0.25或0.55附近，Decoder会倾向于生成像0的图形；在0.45附近，Decoder会倾向于生成像5的图形；在0.6或-0.5附近，Decoder会倾向于生成像3的图形；在0.75附近，Decoder会倾向于生成像8的图形等等。

总而言之，图片的效果和z相关度极大。且根据z的分布来生成图片效果的分布。这符合我们对模型的直觉，因为重建这部分工作是Decoder来进行的，Decoder只需要一个输入z。而Encoder进行的工作是从输入图片中提取并采样得到z。我们在这个探究部分所关心的只是z如何通过Decoder生成重建图片。

c. 将隐层向量z维度设置为2，找出隐层向量的两个维度[-5, 5]值区间内对应的图片生成效果。

我随机采样了100张图片，根据他们的z值进行二维空间排序，最后得到如下结果：



如图所示，两维度处于[-5, 5]值区间内对应的生成图片基本包含了从0-9的所有手写数字。虽然作图效果没有ppt上演示的那么好，但是可以大致看出每个数字各自都有自己所处的概率空间区间。这非常符合我们对模型的预期。

d. 最小化重构误差。

根据我的探究，当z_dim高的时候，相应的重构误差就小。当z_dim达到10及以上的时候，重构误差基本降低到最小了。

实验总结：

本实验提供了最基础的VAE代码，可读性较强，但是整体框架尚需学生搭建，具有较强的挑战性。但是由于网络结构较易搭建，且训练时间短，可完成性还是很强的。不过在没有阅读论文之前，隐层向量维度作为超参数的意义一直让我摸不清头脑。此次实验让我理解了通过拟合正态分布，将图像编码并解码，从而得到新图片的方法，让我理解了一部分图像生成的知识。