一般来说，我们可以认为图像、文字、音频等数据是分布在低维流形空间上的。

GANs通过generator将隐空间的点映射到数据空间中，那么，隐空间的维数怎么选择呢？这是一个值得研究的问题。这里以图像为例进行分析。

首先，隐空间维数不能太低，太低了容易丢失mode，也会产生mode collapse。也就是说，隐空间的维数有个下界，高于这个下界才有可能避免mode丢失的问题。这个下界就是流形的内在维数（intrinsic dimension）。

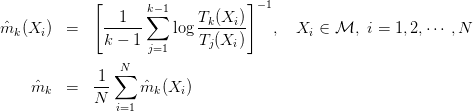
什么叫流形的内在维数呢？下面给出的定义来自于文献[1]。

**(流形定义)\mathbb{R}^n的子集\mathcal{M}称为一个内在维数为m=m(\mathcal{M})，具有p-光滑结构的流形，如果存在一个常数c_p(\mathcal{M})，使得对于任意给定的x \in \mathcal{M}，存在一组m个向量v_1(x), v_2(x), \cdots, v_m(x) \in \mathbb{R}^n，\inf_{\gamma \in \mathbb{R}^m} \|x' - x - \sum_{j=1}^m \gamma_j v_j(x)\| \leq c_p(\mathcal{M}){\|x' - x\|}^{1+p}对任意的x' \in \mathcal{M}都成立。**

简而言之，流形上的点可以用它周围的点逼近。如果对任意点，都可以用m个周围的点进行逼近，那么流形的内在维数就是m。请注意，与向量空间的维数定义不同，向量空间要求基底是固定的，而流形的基底是局部的，因点而异。

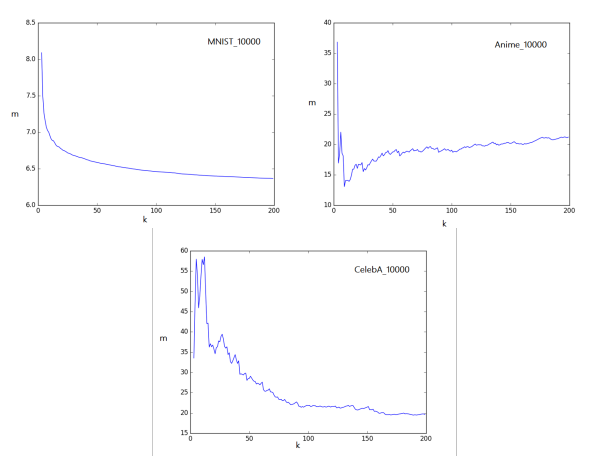
那么，怎么估算内在维数m呢？很多学者提出了数值估算方法，这里介绍一种极大似然估计的方法，详细的推导参看文献[2]。

设我们有一些来自流形\mathcal{M}的样本点\{X_i\}_{i=1}^N，

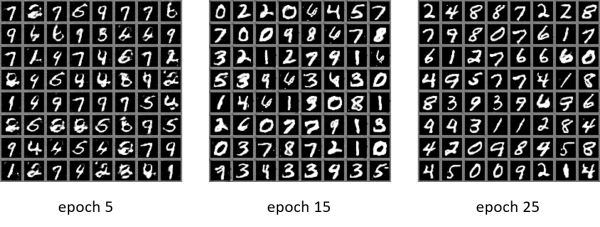


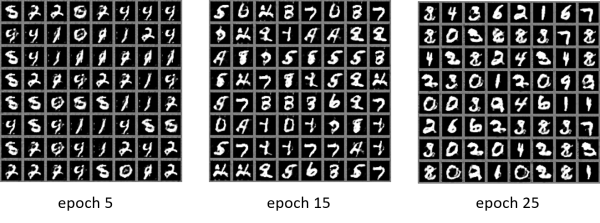
其中，T_k(X_i)表示X_i的k近邻与X_i之间的距离。

利用这个方法，我们可以估算数据集的内在维数。为了节省内存，以下实验结果均为随机采样10000个样本点的计算结果。实验发现，MNIST数据集的内在维数大约为6.5，动漫数据集的内在维数大约为21，而CelebA数据集的大约为20。需要注意的是，数据集的样本点可能远不止这么低内在维数，但是由于某些mode样本很少，例如动漫数据集的背景几乎每张图像都不一样，这些mode会被忽略，对内在维数没有贡献。



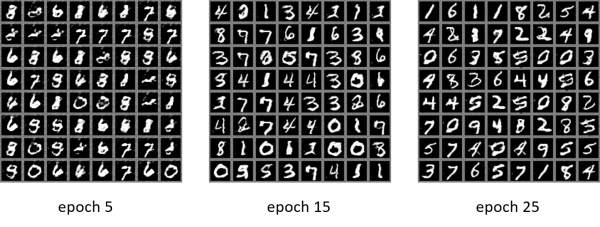
实验中我们对隐空间的选择不应低于这些下界。以MNIST为例，选择隐空间维数为10，能得到不错的效果。可以看到，生成的图像具有数字类型、倾斜角度等变化，它们都是隐空间内在维度的一部分。

如果选择隐空间维数低于内在维数6.5，如选择1，我们得到如下的结果。可以看到，一维的隐空间控制的是生成的数字类型，在同一个epoch内，生成的相同的数字只会有一种形态。不同的epoch之间，相同数字形态可以不同。

PS：隐空间设置为1维，生成的图像质量不行，原因在于隐空间的这个维度编码的不只是数字类型，它应该还有包含其他信息。数字类型是离散的，测度为0，不会充满整一维的空间，而其他信息一般是连续的，测度大于0，多种信息交织不完整地编码在一维空间中，导致生成图像与真实图像有一定差距。

其次，隐空间维数也不能太高，太高了占用太多显存，模型更强，搜索空间更大，存在更多的鞍点，学的越慢。

下图为MNIST数据集隐空间选择为2048维(2048>28\*28)的结果。可以看到，epoch 25时，学得差不多了，每个数字的形态比隐空间为10的情况要丰富很多。隐空间维数越高，更多罕见的mode会对内在维度做出贡献。隐空间的选择是否有一个合理的上界，还需要进一步研究。



参考文献

1. Yu K, Zhang T, Gong Y. Nonlinear learning using local coordinate coding[C]//Advances in neural information processing systems. 2009: 2223-2231.

2. Levina E, Bickel P J. Maximum likelihood estimation of intrinsic dimension[J]. Ann Arbor MI, 2004, 48109: 1092.