流形空间分析数据增强

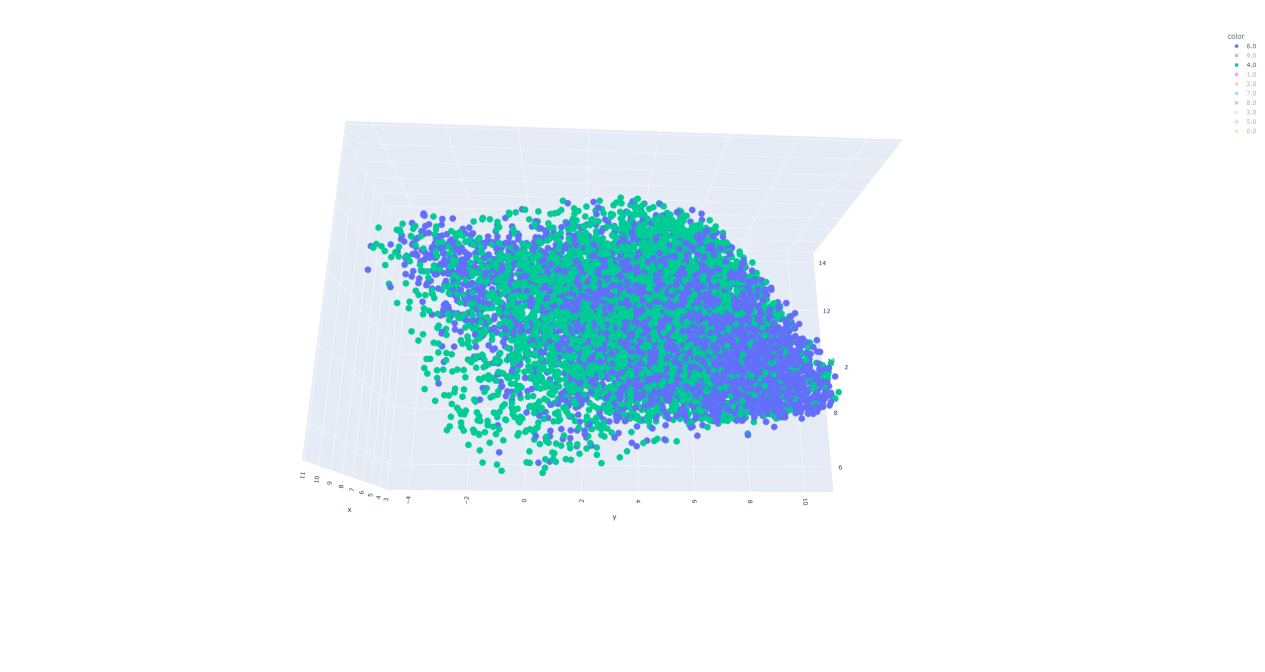
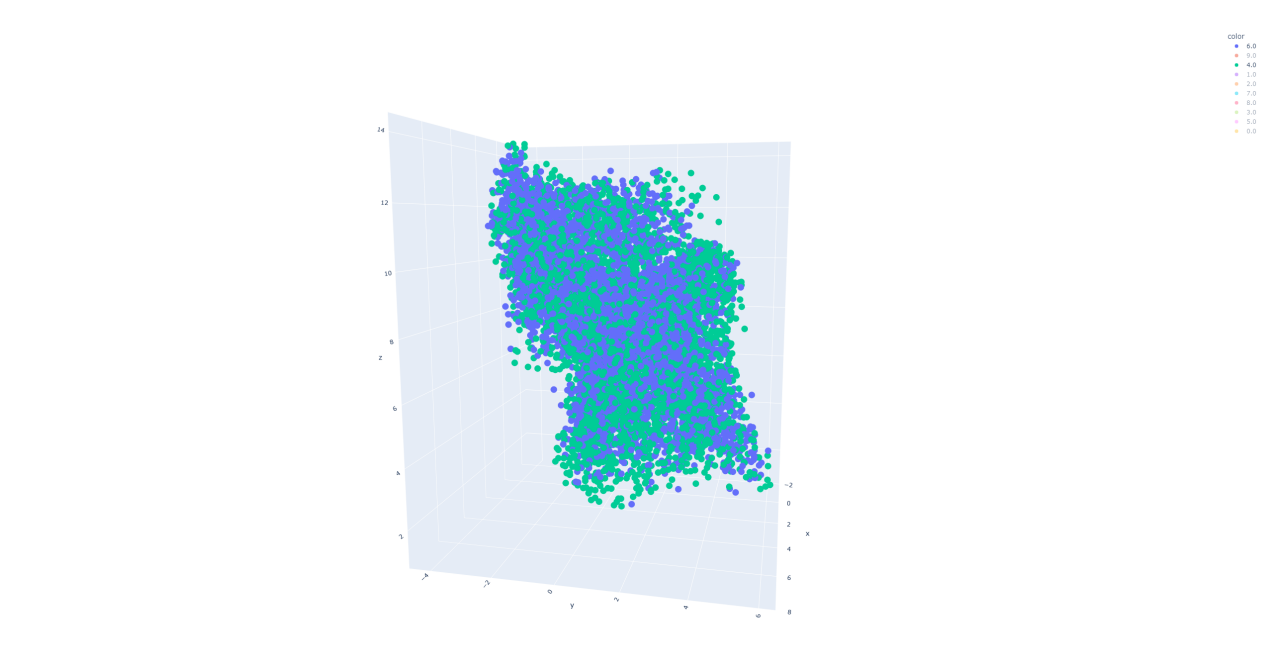
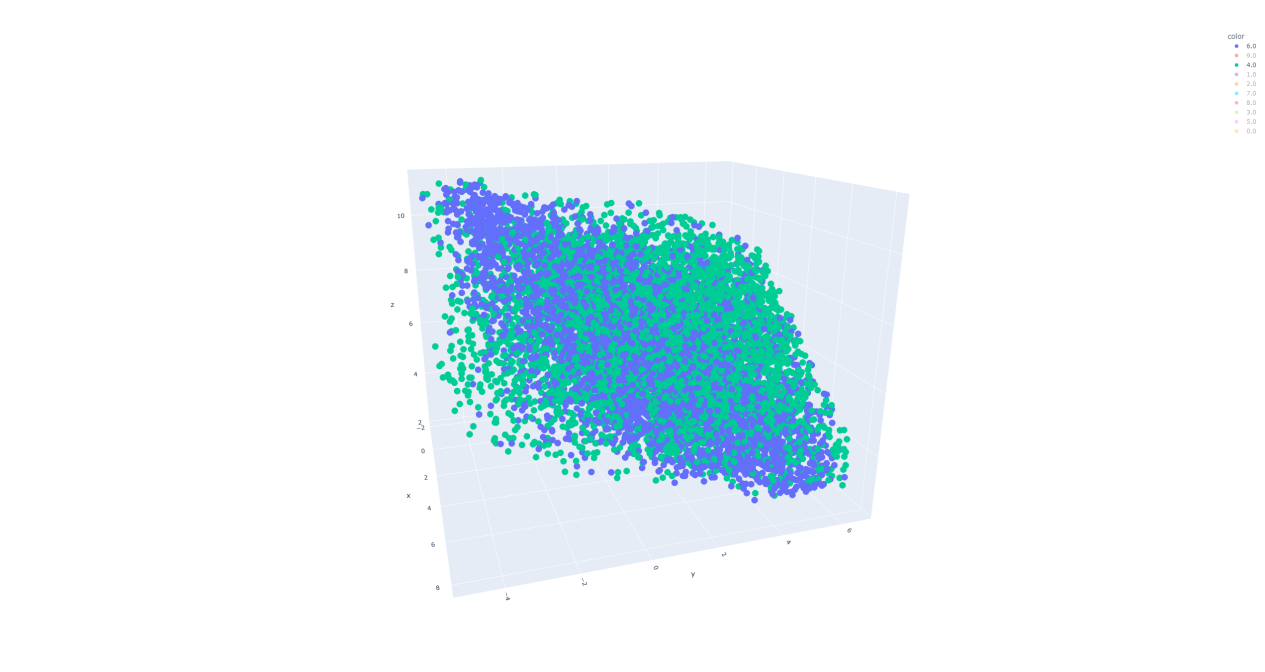
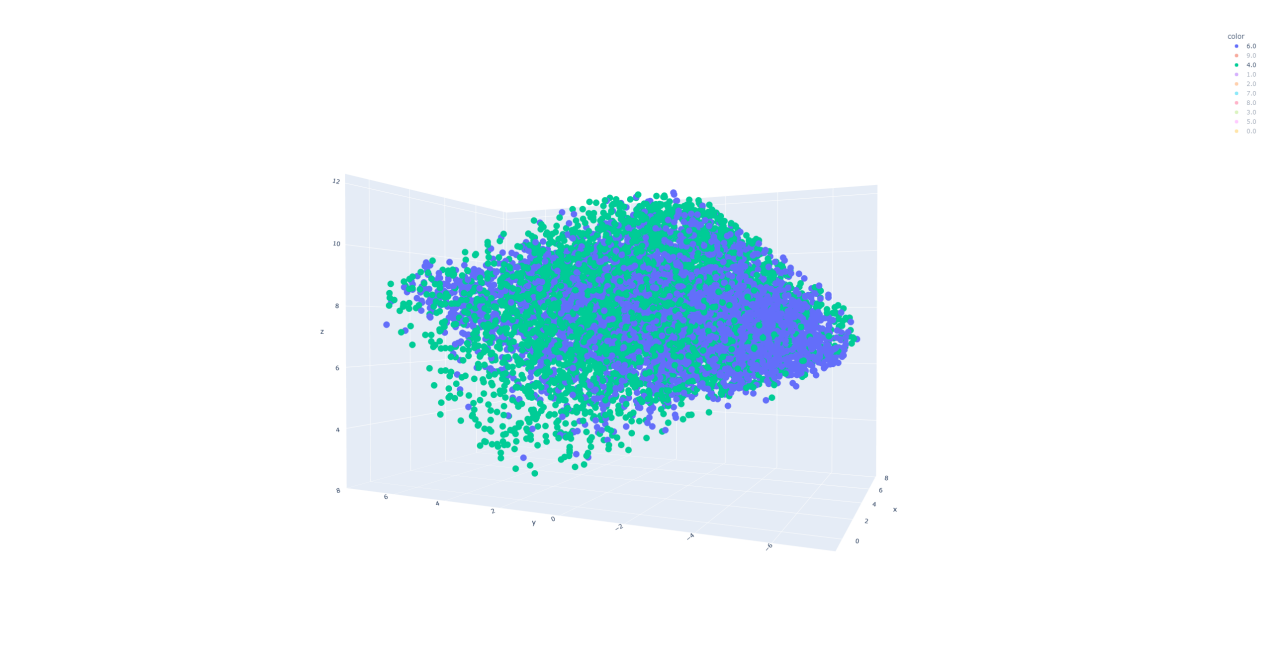
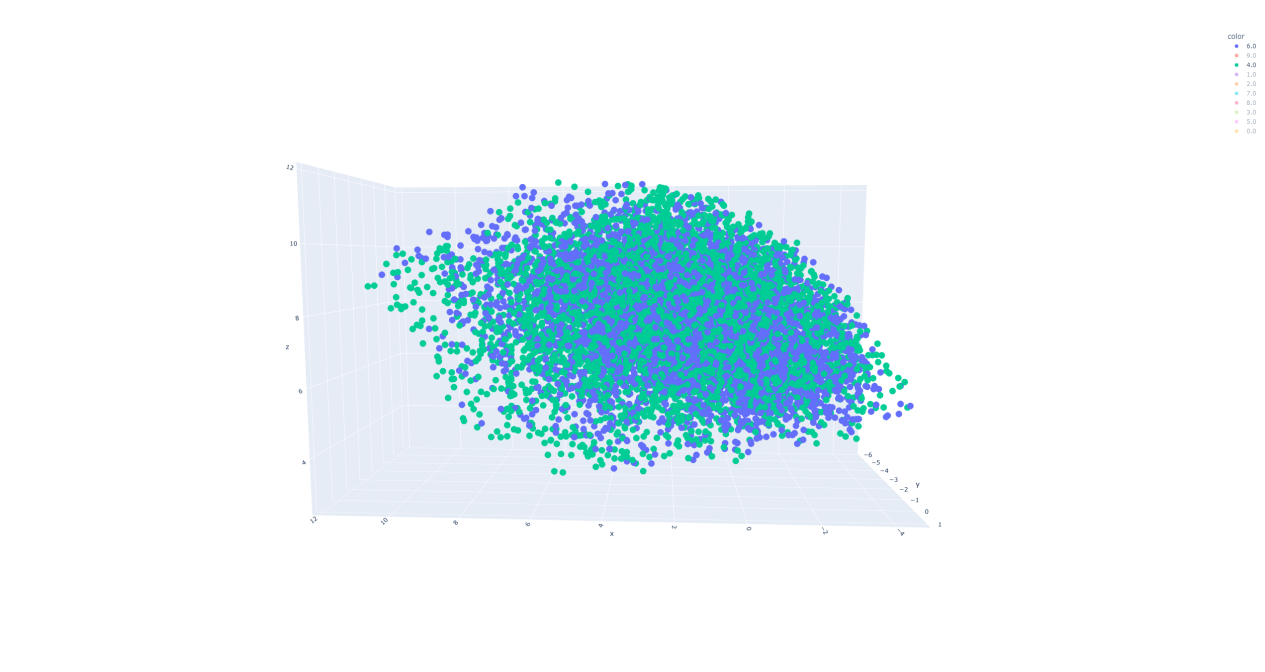
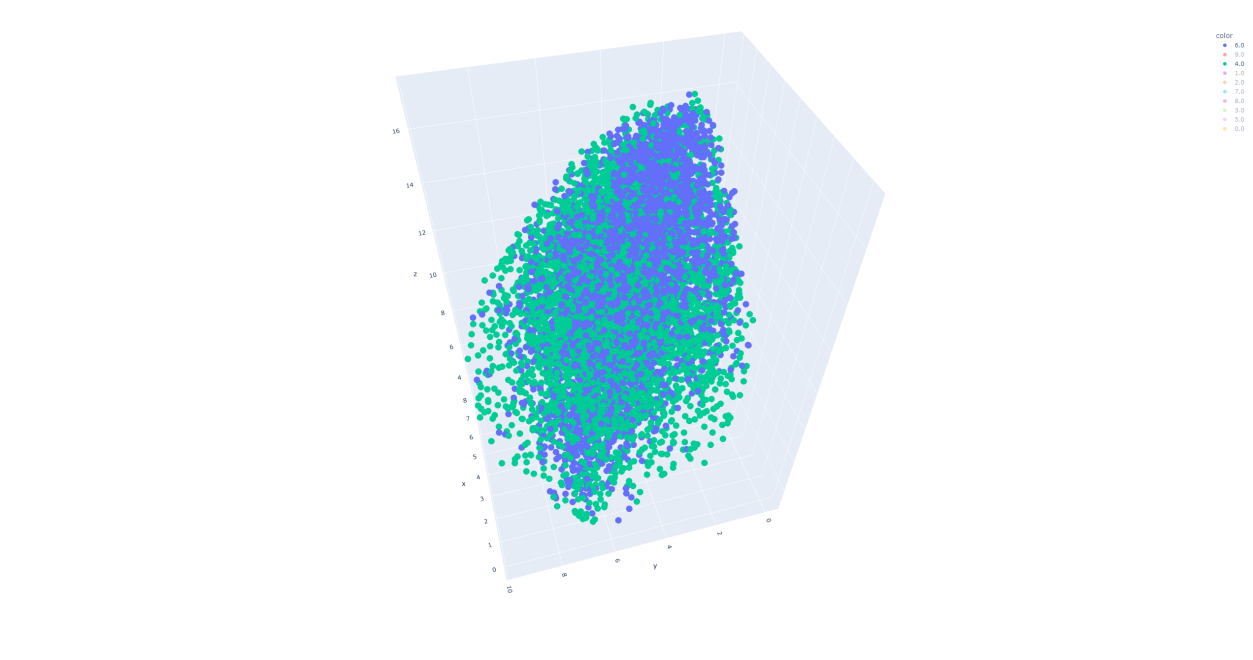
实验Case1

1. 实验的目标

在流形空间中分析数据增强对数据的影响及为什么能改善神经网络.

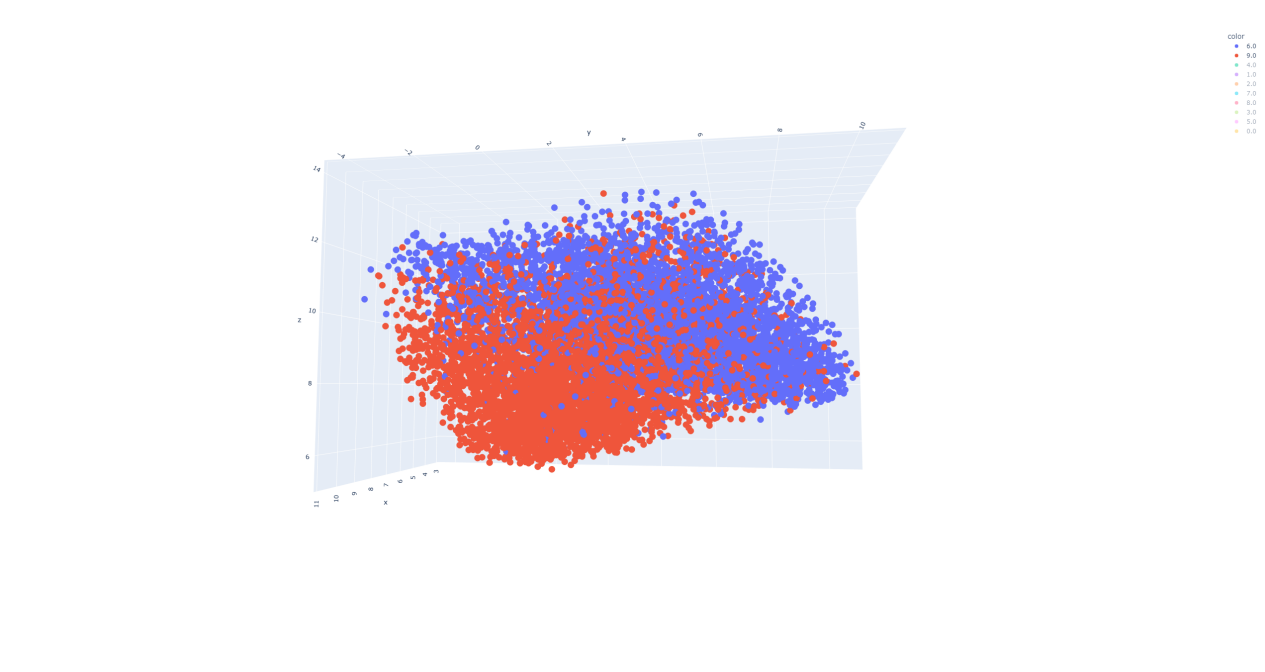
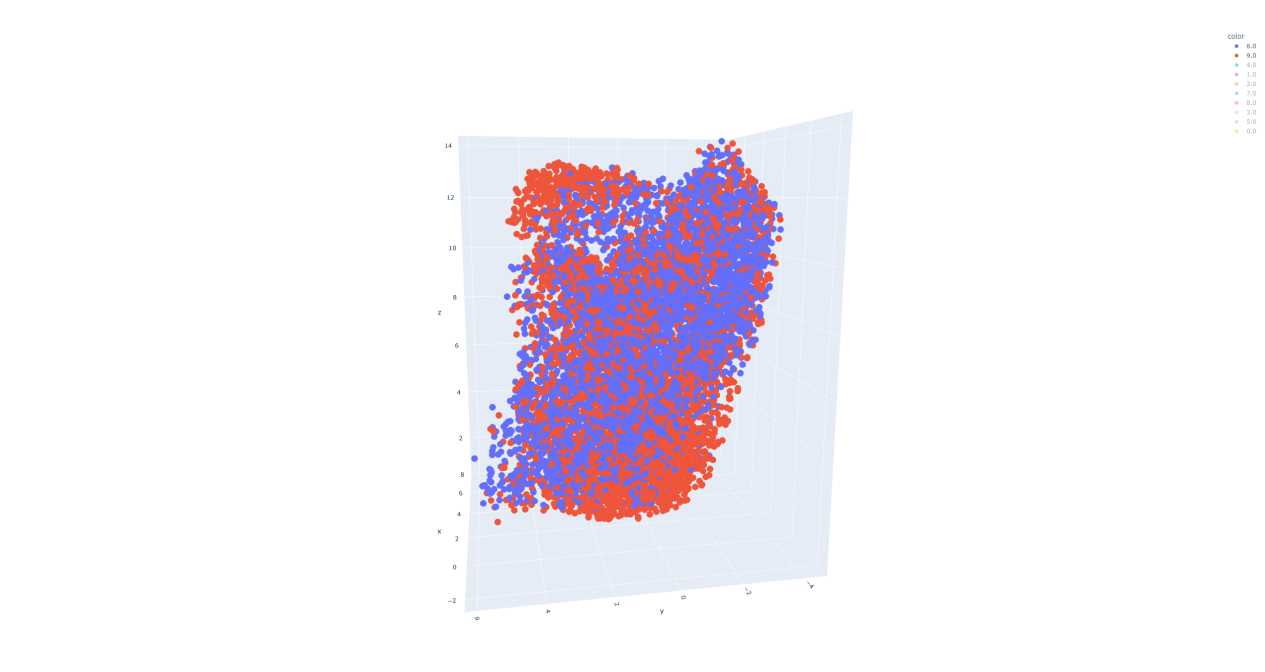
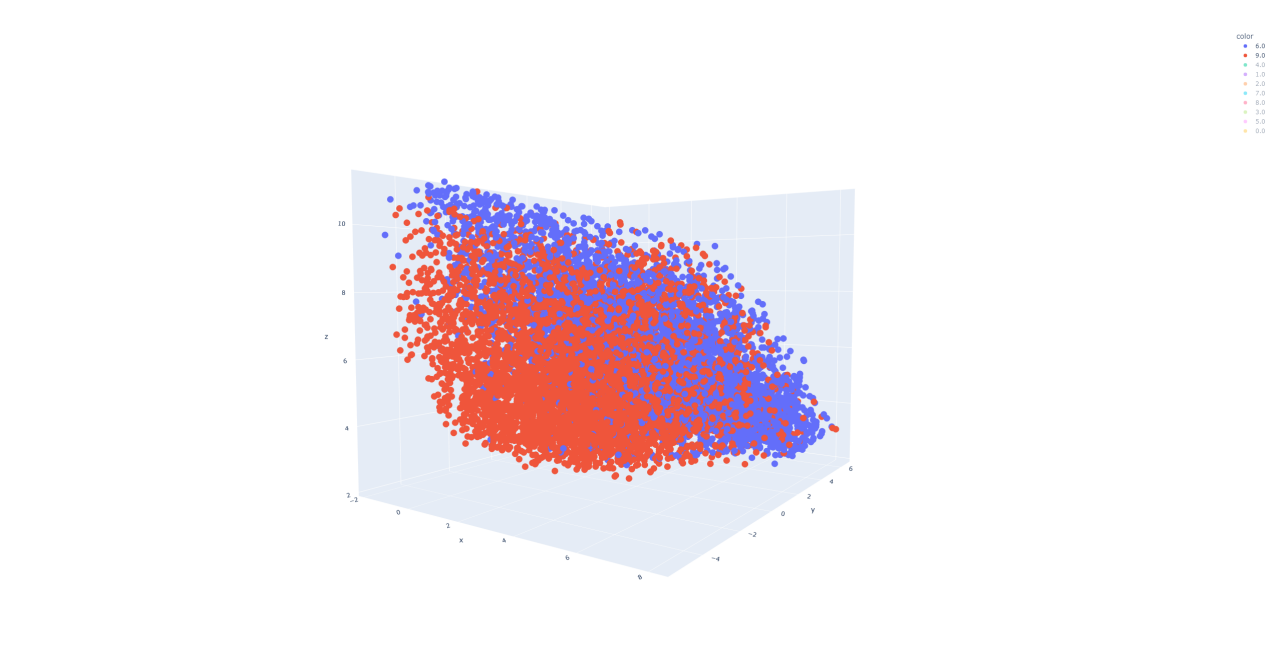
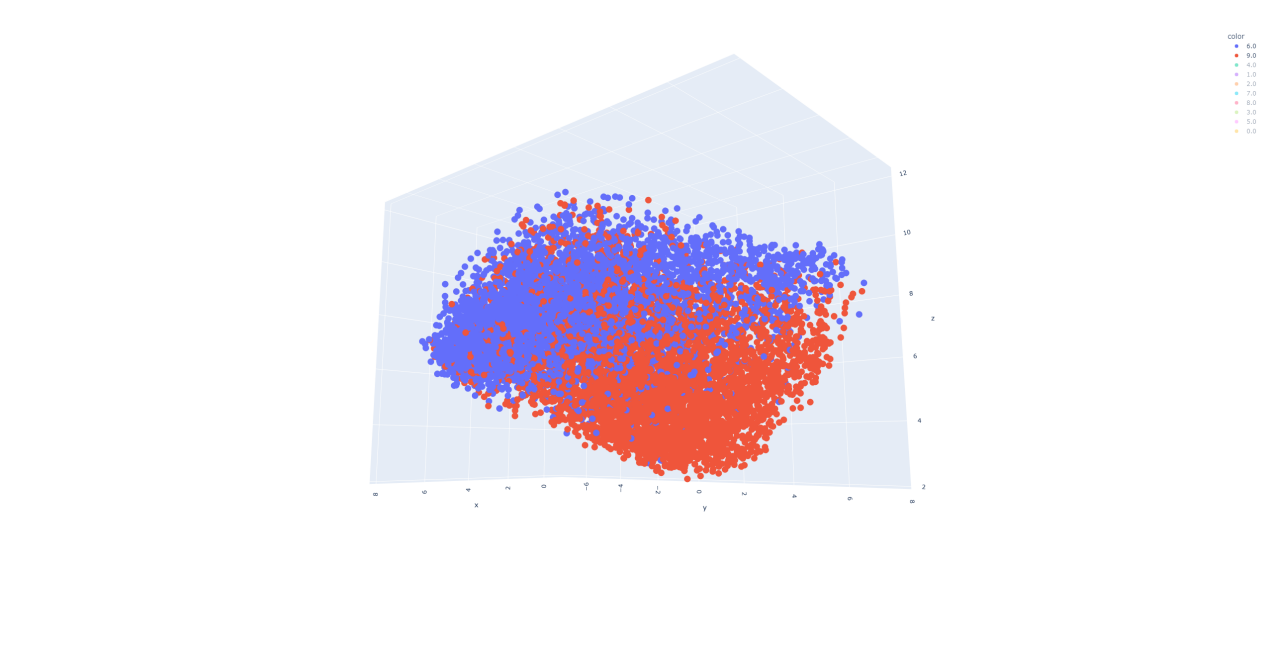
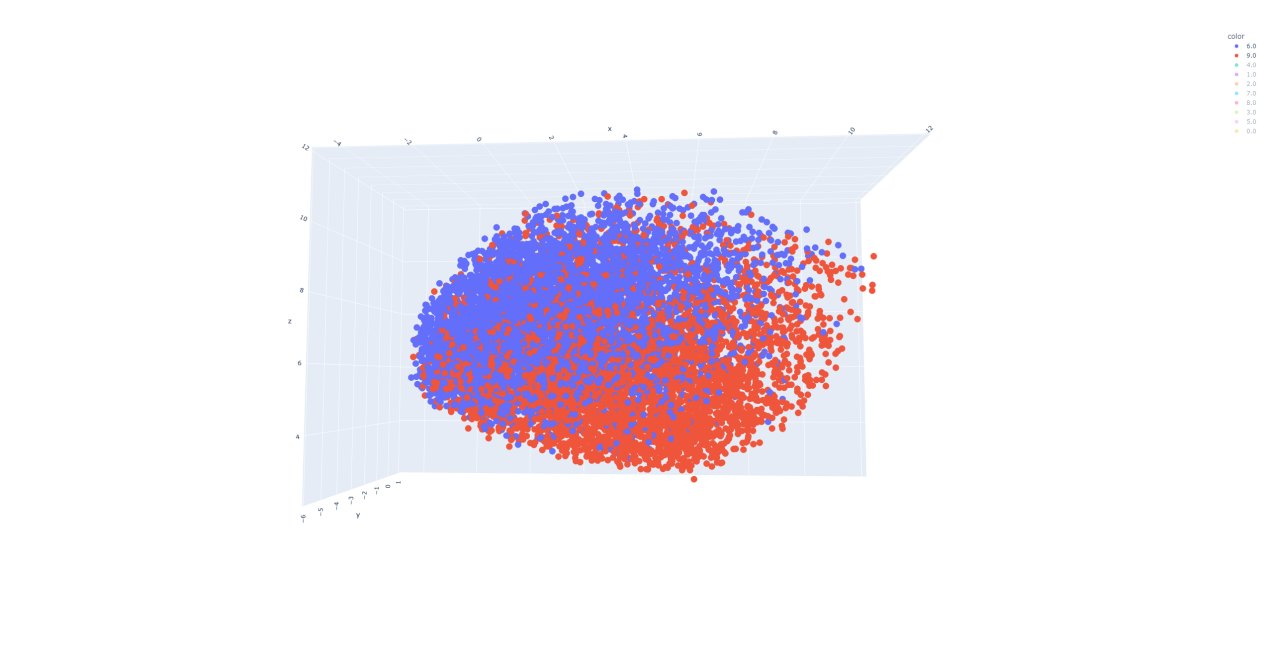
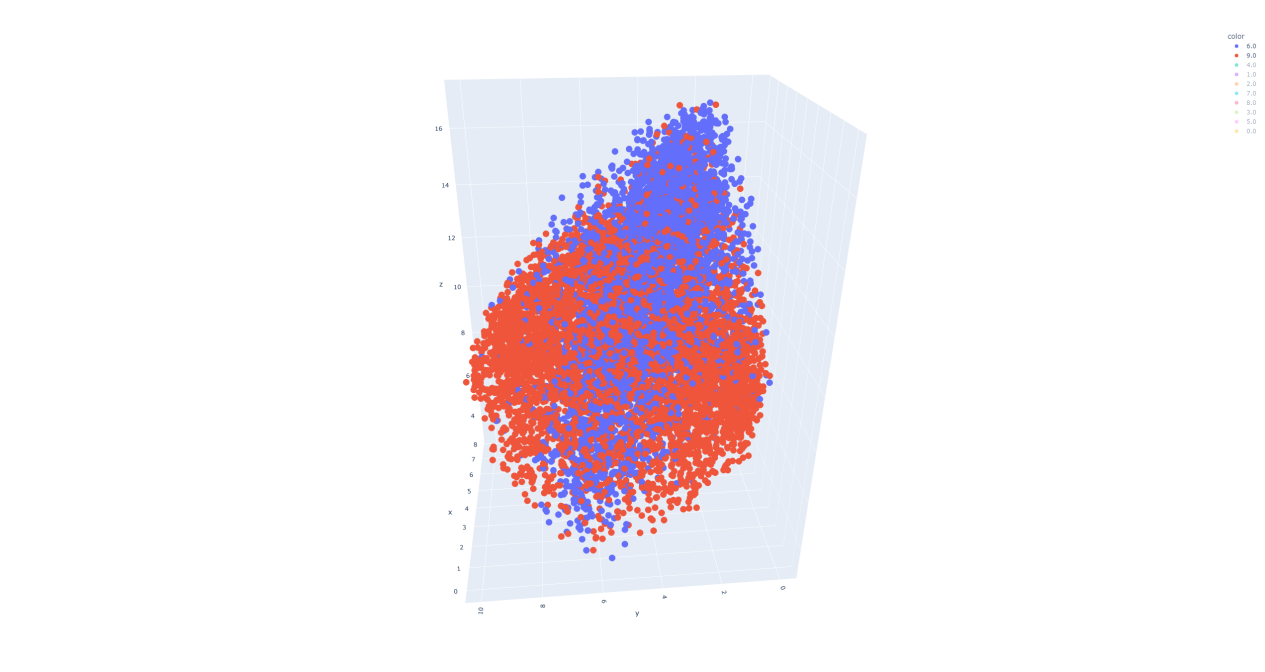
1. 实验的设计
   1. 实验流程设计
      1. 使用工具python,主要的第三方库:panda,torch,torchvision,plotly,numpy,umap,数据集为cifar10和cifar100,用panda把数据转化得易于降维,然后用umap降维算法对其降维并用plotly将其绘制,数据增强一部分来自于torchvision.
   2. 工具
      1. Panda
      2. torch
      3. torchvision
      4. Plotly
      5. numpy
      6. umap
   3. 外部的主要参数
      1. 降维算法的参数
         1. local\_connectivity=2
         2. random\_state=42
         3. min\_dist=0.99
         4. n\_neighbors=1000
         5. n\_components=3
         6. 其他参数均为默认
      2. 数据增强参数
         1. RandomResizedCrop
            1. size=32
         2. Normalize
            1. mean=[0.486,0.456,0.406]
            2. std=[0.229,0.224,0.225]
         3. 其他均为默认
2. 实验的数据和表格

数据繁多,不易分析,挑取其中特殊的几组进行展示并分析.



注:此图为cifar10中的4,6组(从1计数),及猫和狗.图1-5采用的数据增强方式为1.RandomVerticalFlip,RandomHorizontalFlip,2.RandomResizedCrop,3.ColorJitter,4.Grayscale,5.Autoaugmentation,图6为无数据增强的原始图.以后若无特殊说明,图片顺序表示的意义均与此相同.

可以看到,在cifar10中,用umap降维的结果中猫和狗的分布几乎是相同的,只有细微的分布上的侧重点有差异,如在第六张图(无数据增强)中,狗的分布成纺锤形,而猫更趋向平行四边形.但总的来说,它们分布几乎是相同的.事实上,在cifar10数据集中,猫和狗的差异点相较与其他任何两组来说都更小,如:



注:为6,9组,即狗和轮船.

可以看到差距比较大的两组分布是几乎可以用一条线将其区分的.但采用了Autoaugmentation(自动数据增强)的图像比较特殊.整体上讲较其他组而言变得更加混淆了.

4.实验结果分析

对实验结果的分析基于以下理论:

就cifar10而言,它的总的数据在高维中具有某种几何形状,及集中在某个低维的流形附近,事实上,我们降维的结果--三维应该只是其完整流形的投影,因为cifar10的自由度肯定不止三维,但是,即使是真正流形的投影,其形状应该也是相对确定的.然后对每一类数据而言,它自身所成的流形(或者我们看到的流形投影)应该也是相对确定的.于是我们试图探究每一类数据自身流形的变化和它相对于整体的结构变化对神经网络训练结果的影响.

算法分析:

因为数据的流形空间是相对确定的,所以神经网络的训练实际上(也许)是对其流形空间的概括,当训练数据量一定时,我们希望用同样的数据量将其囊括的流形空间尽可能的表达,对这个问题进行简化,就是有一个未知的函数,点的数目一定时,点怎样分布最能表达出这个函数,很明显,函数都有自己的特殊点(如极值,拐点等),但在流行空间中,我们应该是不知道这个流形的特殊点的,所以,还有一种分布,就是尽可能均匀的分布可以帮助我们最大程度上去模拟出那个未知的函数,也是那个不知道形状的流形.

所以,当无法知道流形的”特殊点”时我们可以通过衡量流形的分布均匀程度来判断数据增强的优劣(事实上,在训练过程中,因为每次训练轮次都会调用一次数据增强,所以带有随机性的数据增强方法可以用同样的数据去模拟出更多的点,即让点分布得更密集,来帮助神经网络更好的训练,而我们受限于内存,只能分析一个数据增强的降维流形图).所以,衡量流形自身的变化的方法可以通过衡量其样本相对于自身形状的分布均匀程度来得到.

想法1:

以每个点为球心,作半径不定的球,使所有这样的球相切,最后计算半径的方差.(这个理论上是能够衡量出分布均匀程度的,但是...这个状态我不知道是不是都存在,存在又是否唯一,也不知道怎么去在数学上找到这个状态)

想法2:

回到简化为那个未知函数的问题,衡量一个函数是否分布均匀事实上只用衡量它自变量的均匀就可以,所以可以把流形空间中每一个点集(一个类)分别视为因变量为x,y,z的函数,然后计算对应自变量的协方差(更多变量的方差?),最后求平均值作为结果.(感觉这样做在流形空间里好像没有意义)

新算法:

对每一类,分别对x,y,z轴的变量排序后计算距离间方差的均值.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 无数据增强（origin） | RandomVerticalFlip RandomHorizontalFlip | RandomResizedCrop | ColorJitter | Grayscale | Autoaugmentation |  |
| 0.592/0.313 | 0.620/0.338 | 0.692/0.358 | 0.574/0.304 | 0.487/0.214 | 0.673/0.323 |  |
| 99.5414378231827, | 100.73 | 108.08 | 99.54 | 108.45 | 108.60 |  |

注:行一为准确性/泛化性,行二为流形不均匀度的计算结果(结果越大越不均匀)

结果分析:对于准确性和泛化性都变好的1,2,5组来说,其不均匀度都是在增大的,而准确性和泛化性都变差的3,4组来说,3组不均匀度减小,4组增大了.数据变得更不均匀(或者说,更有重点?)是数据增强的必要条件之一?

还有一点点值得注意,没有数据增强的和ColorJitter(改变图片亮度,对比度等人眼因素)对流形下数据的不均匀度几乎没有影响,(ColorJitter结果是否变差可以等更多模型训练出结果后再说),甚至降维后的数值结果也是不相同的,但其不均匀度却几乎没有差别.有待cifar100的数据.