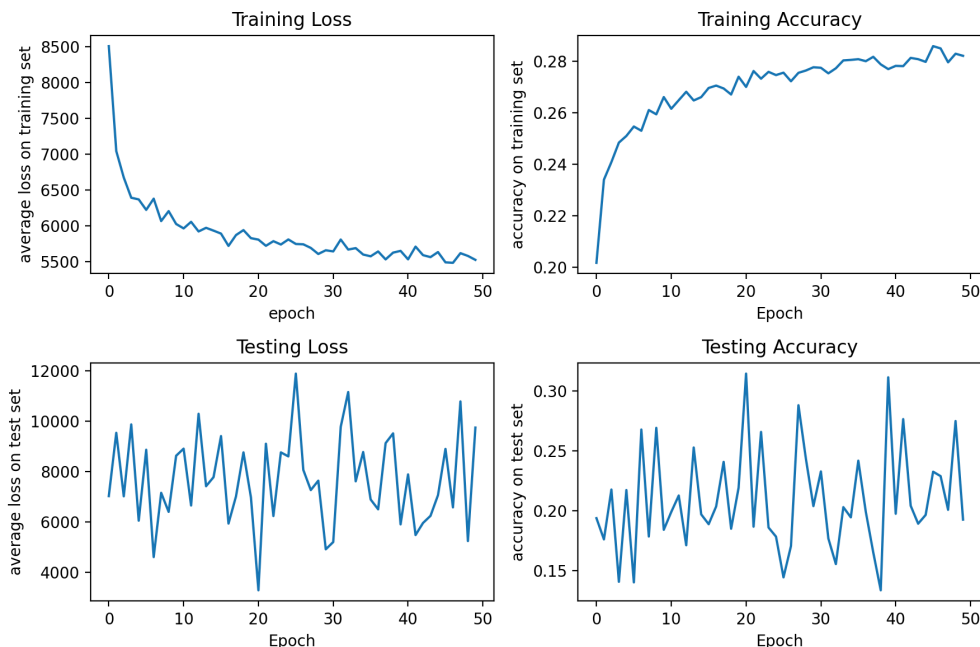


机器学习与数据挖掘——作业 2

林宇浩 21311274

一、线性分类器(Softmax 分类器)

训练结果如下图所示：



可以看到，线性分类器在此任务上的效果并不理想，尽管随着训练轮数的增加，训练损失在不断下降，在训练集上的准确率也有不断上升，但是测试损失并没有随着平稳下降，在测试集上的准确率也并没有随着平稳上升，测试中两个数据都在剧烈波动。这说明模型的拟合能力太差，没有学习到全面的特征，导致从曲线中看仿佛是在随机拟合。

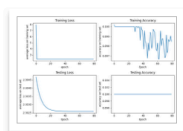
同时注意到，尽管训练损失在下降，但是训练损失的绝对值还是非常大，下降到 5500 左右后已难以继续下降，这也说明了线性分类器在此任务中欠拟合了。

二、多层感知机(MLP)

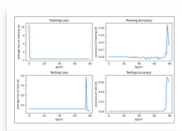
1、两层感知机

两层感知机的结构为：1 个输入层（维度与输入数据维度对应）、1 个隐藏层、1 个输出层（神经元数量与输出类别数量对应）

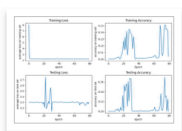
具体训练数据详见附件中文件夹 2_layers_MLP，图片名称对应隐藏层中神经元数量：



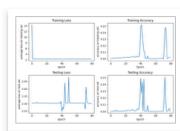
16.png



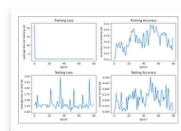
32.png



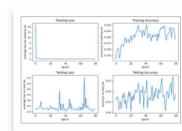
64.png



128.png



256.png



512.png

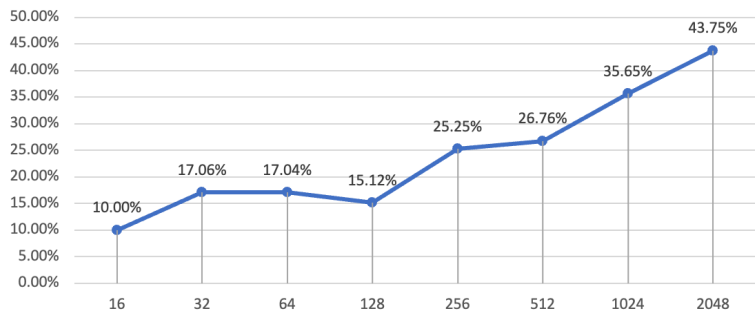
训练日志文件为 output_log_2_layers_MLP.txt：

```
Epoch[1/80] Average Loss:7.9526, Train Accuracy:10.02%, Train Time:1.70s, Test Average Loss:2.3047, Test Accuracy:10.00%, testing_time:2.49s
Epoch[2/80] Average Loss:2.3045, Train Accuracy:10.00%, Train Time:1.20s, Test Average Loss:2.3044, Test Accuracy:10.00%, testing_time:2.48s
Epoch[3/80] Average Loss:2.3042, Train Accuracy:10.00%, Train Time:1.20s, Test Average Loss:2.3041, Test Accuracy:10.00%, testing_time:2.45s
Epoch[4/80] Average Loss:2.3040, Train Accuracy:10.00%, Train Time:1.21s, Test Average Loss:2.3039, Test Accuracy:10.00%, testing_time:2.44s
Epoch[5/80] Average Loss:2.3038, Train Accuracy:10.00%, Train Time:1.20s, Test Average Loss:2.3037, Test Accuracy:10.00%, testing_time:2.47s
Epoch[6/80] Average Loss:2.3036, Train Accuracy:10.00%, Train Time:1.19s, Test Average Loss:2.3036, Test Accuracy:10.00%, testing_time:2.46s
```

两层感知机训练过程为，为对比不同隐藏层神经元数的影响，将隐藏层神经元数作为唯一变量，对不同的模型均训练 80 轮，观察其训练数据的变化。

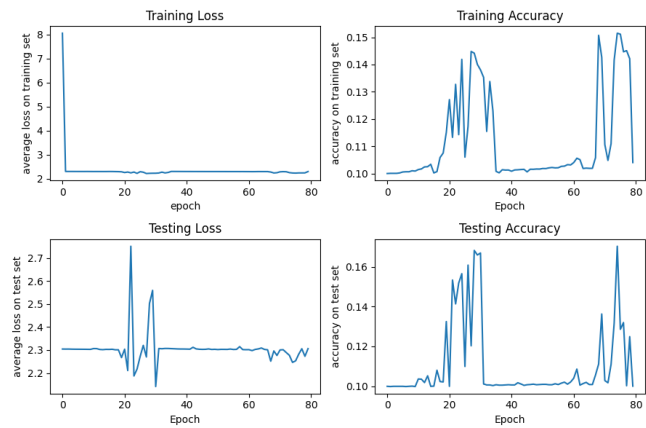
下表中，第一行为隐藏层的神经元数量。第二行为 80 轮（epoch）训练中，测试集上准确率的最大值。

神经元数	16	32	64	128	256	512	1024	2048
准确率	10.0%	17.06%	17.04%	15.12%	25.25%	26.76%	35.65%	43.75%

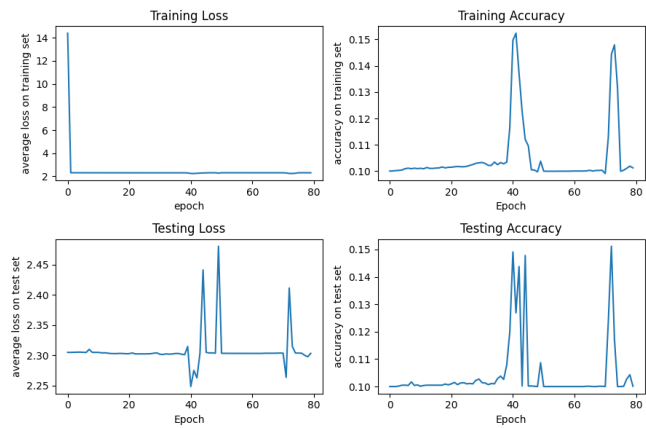


可以看到，对于只有两层的感知机，在隐藏层维度小于输入层维度（ $32 \times 32 \times 3 = 3072$ ）的情况下，总体上看隐藏层神经元数量越多，其测试准确率越高，可能原因是由于层数比较少，且数据量比较大，较多的神经元能够帮助其弥补拟合能力上的不足。

(1) 隐藏层的神经元数量为 64：



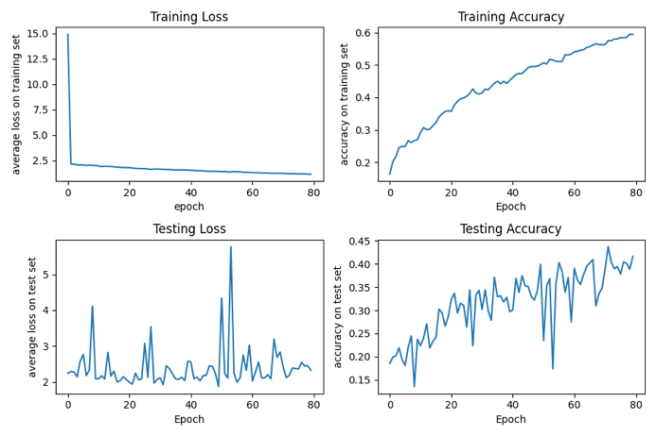
(2) 隐藏层的神经元数量为 128：



(3) 隐藏层的神经元数量为 1024：



(4) 隐藏层的神经元数量为 2048：



从上面 4 幅图的训练数据也可以看出，对于较少的隐藏层神经元数量，其不就后即出现了过拟合，之后又在过拟合后重新尝试正确拟合，所以准确率会存在两个峰。同时可以看到，其训练损失虽然比较低，但是不久后就难以继续下降了。而对于较多的隐藏层神经元数量，其训练损失直到 80 轮还是在不断下降的，测试准确率随着训练准确率的上升而上升。从坐标轴可以看到，图 4 的训练损失是最大的，然而其测试和训练准确率反而是最好的，这也说明了训练损失并不是越小越好。

2、三层感知机

三层感知机的结构为：1 个输入层（维度与输入数据维度对应）、2 个隐藏层、1 个输出层（神经元数量与输出类别数量对应）

具体训练数据详见附件中文件夹 3_layers_MLP

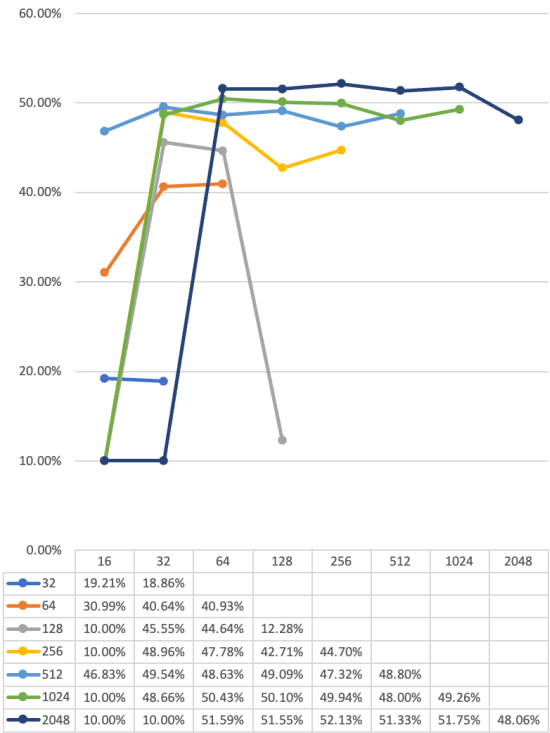
下表中，纵坐标为第 1 层隐藏层的神经元数量。横坐标为第 2 层隐藏层的神经元数量。表格中的数据为 80 轮（epoch）训练中，测试集上准确率的最大值。

	16	32	64	128	256	512	1024	2048
32	19.21%	18.86%						
64	30.99%	40.64%	40.93%					
128	10.0%	45.55%	44.64%	12.28%				
256	10.0%	48.96%	47.78%	42.71%	44.7%			
512	46.83%	49.54%	48.63%	49.09%	47.32%	48.8%		
1024	10.0%	48.66%	50.43%	50.1%	49.94%	48.0%	49.26%	
2048	10.0%	10.0%	51.59%	51.55%	52.13%	51.33%	51.75%	48.06%

从上图中可以看出，对于三层感知机，在第 1 层的维度小于输入层维度（32*32*3=3072）的情况下，第 1 层隐藏层的神经元数量越多，测试准确率越高。对于第 2 层，从右图的趋势中可以看出，在第 2 层的维度不超过第 1 层维度的情况下，随着第 2 层隐藏层神经元数量的增加，测试准确率的变化趋势为先上升后下降。下表展示了第 1 层神经元数量固定时，第 2 层神经元数量为多少能够使得测试准确率最大：

第 1 层神经元数量	使得测试准确率最大的第 2 层神经元数量
32	16
64	64
128	32
256	32
512	32
1024	64
2048	64

整体来看，对于神经元数量的选择，第 2 层应该比第 1 层数量要少，第 2 层神经元数量的增长速度比第一层缓慢很多。



3、四层感知机

使用上面效果较好的几个三层感知机，再增加一层隐藏层，观察第 3 层隐藏神经元数量的影响，数据如下表所示：

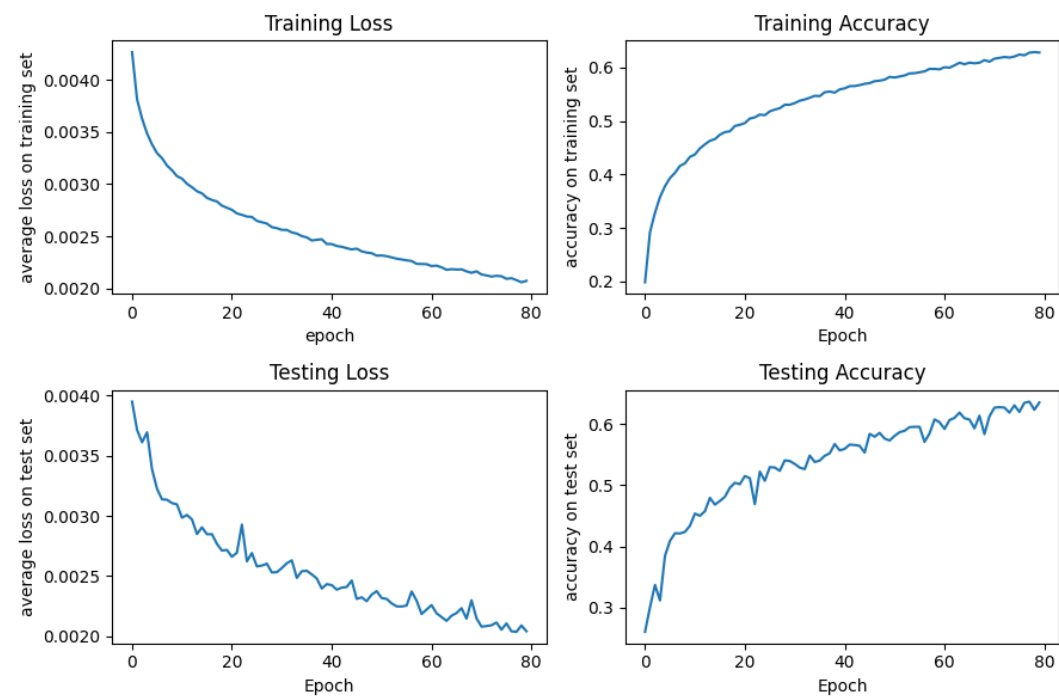
第 1 层神经元数量	第 2 层神经元数量	第 3 层神经元数量			没有第 3 层时
		16	32	64	
256	32	10.01%	48.72%	49.16%	48.96%
512	32	66.94%	71.84%	74.69%	49.54%
1024	64	10.00%	80.54%	86.56%	50.43%
2048	64	10.02%	87.48%	91.43%	51.59%

可以看到，第 1 层神经元数量越多时，添加第 3 层神经元后带来的提升越明显。反之，如果前两层隐藏层中神经元的数量比较少，添加第 3 层神经元带来的提升并不显著，甚至还可能下降。可以看到，上表中最大测试准确率都出现在第 3 层神经元数量为 64 时，即使第 3 层比第 2 层神经元数量多的情况下也如此，主要原因可能是在没有

第 3 层时，各个模型的准确率本身就不高，即使训练了 80 轮，训练集上的准确率也只有百分之七八十的水平，模型的主要瓶颈还是欠拟合而不是过拟合，所以第 3 层参数越多效果越好，但是巨大参数量带来的问题也是很明显的，其训练耗时显著增加。

三、卷积神经网络 CNN

下图为 LeNet 的训练数据：



接下来以 LeNet 为基础，探究不同模型结构因素的影响。

1、卷积层数

卷积层层数	各层卷积核大小	各层滤波器（卷积核）数量	测试准确率
1	5	16	71.35%
2 (LeNet)	5、5	16、32	63.50%
3	5、3、3	16、32、64	57.65%

可以看到在本次任务中，1 层的效果是最好的，主要原因可能是本次任务的图片大小比较小，只有 32*32，如果经过太多次卷积和池化，特征图的大小将过于小，比如 3 层卷积中，最后特征图的大小只有 2*2，这导致能够说明的特征信息比较有限，所以层数越多反而准确率越差。

2、滤波器数

第 1 层滤波器数量	第 2 层滤波器数量	测试准确率
8	8	49.96%
8	16	55.03%
16	16	56.69%
16	32	61.68%
32	32	64.04%
32	64	68.79%
64	64	71.49%
64	128	76.83%

可以看到，在本次任务中，滤波器的数量越多，测试准确率越高，可能的原因应该仍然是当前准确率还比较低，

模型的主要瓶颈是欠拟合而不是过拟合，所以通过增加滤波器数量增加参数能够提高模型的拟合能力，从而使得参数越多准确率越高。

3、Pooling 的使用

第 1 层卷积后是否使用 Pooling	第 2 层卷积后是否使用 Pooling	测试准确率
Yes	Yes	63.50%
Yes	No	68.32%
No	Yes	77.00%
No	No	93.15%

可以看到，不使用池化层准确率最高，主要原因应当仍然与之前类似，由于本次任务的图片大小只有 32*32，而池化操作会导致特征图大小直接减半，池化操作一般的目的是减小特征图的空间尺寸，从而降低计算量和参数数量，然而在本次任务中特征图已经较小，使用池化操作反而会导致特征信息的丢失，从而使得准确率降低。

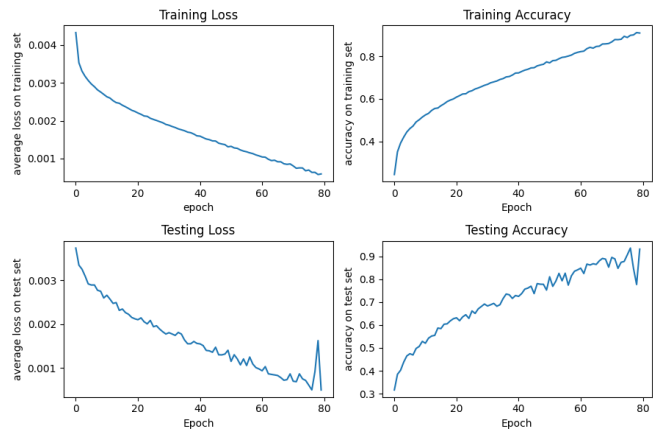
四、比较 SGD 算法、SGD Momentum 算法和 Adam 算法

模型	SGD 算法	SGD Momentum 算法	Adam 算法
LeNet 去掉第 1 层 Pooling 层	77.00%	100.00%	99.62%
LeNet 去掉全部 2 层 Pooling 层	93.15%	100.00%	99.65%
两层感知机（隐藏层神经元数量：1024）	35.65%	10.38%	10.00%
三层感知机（隐藏层神经元数量：256、32）	48.96%	10.00%	10.00%
四层感知机（隐藏层神经元数量：512、32、32）	71.84%	10.00%	10.00%

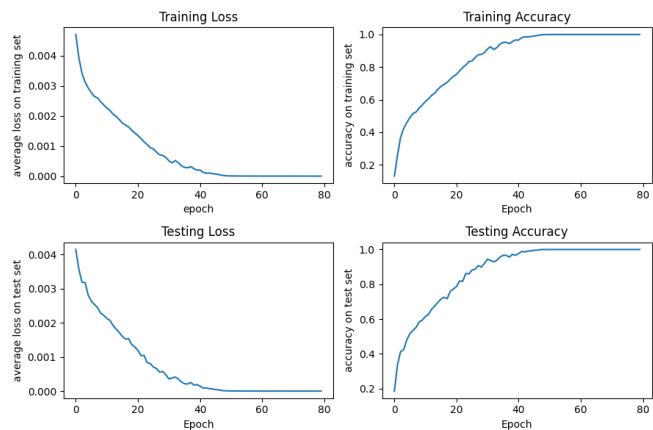
可以看到 SGD Momentum 算法和 Adam 算法对 LeNet 的优化效果非常良好，性能得到了大幅提升，但是其对多层感知机并不适用。

对 LeNet 去掉全部 2 层 Pooling 层后得到的模型进行分析：

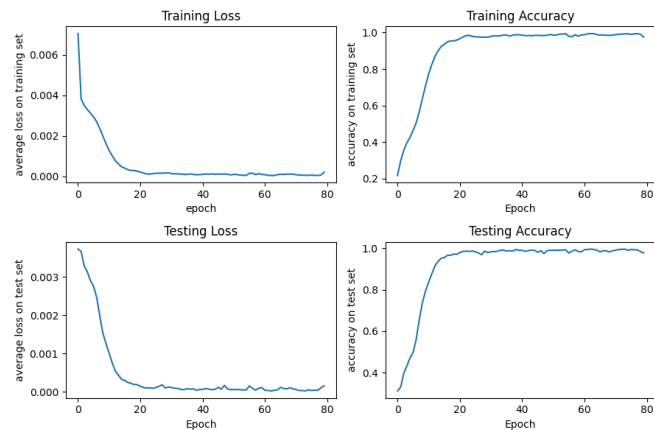
(1) SGD 算法：



(2) SGD Momentum 算法：



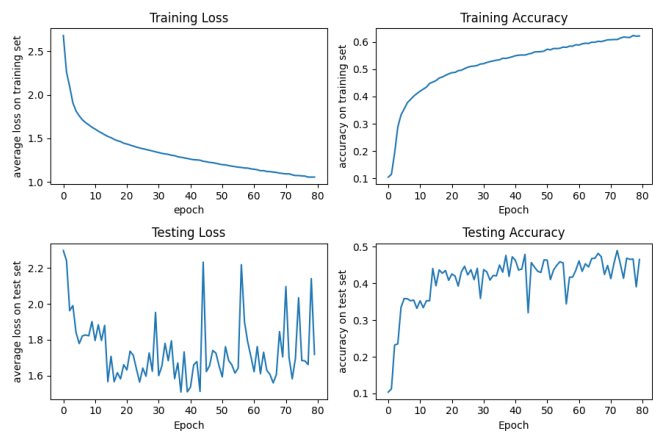
(3) Adam 算法：



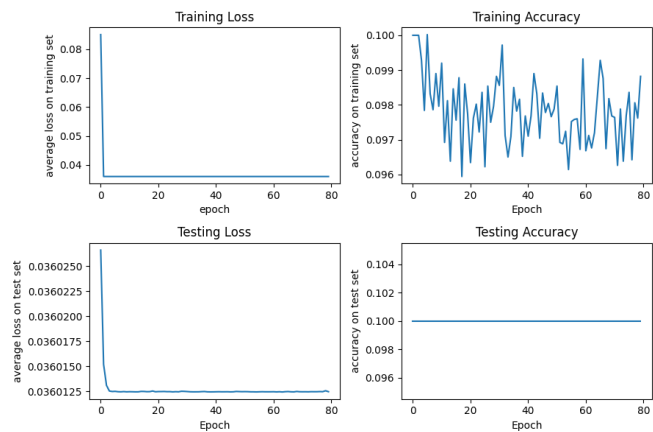
可以看到，使用 SGD Momentum 算法后，模型的收敛速度变得更快，而且测试损失和测试准确率曲线变化得更加平滑，不会像在 SGD 中那样剧烈波动。使用 Adam 算法后，收敛速度跟进一步变快，不到 20 轮训练准确率就达到 90%以上，比 SGD Momentum 算法快了一倍。

然后对三层感知机模型的训练进行分析：

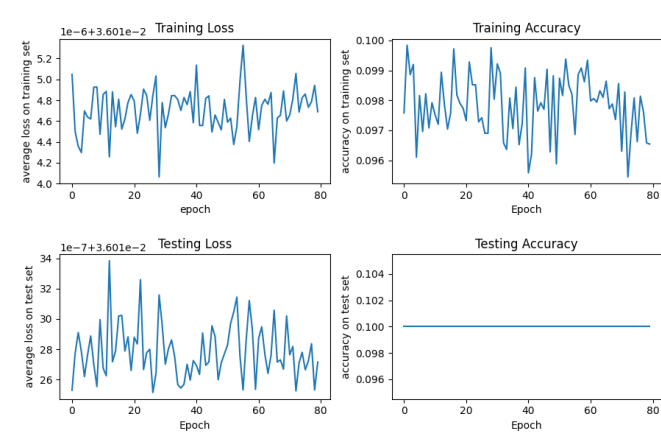
(1) SGD 算法：



(2) SGD Momentum 算法：



(3) Adam 算法：



可以看到在 SGD Momentum 算法和 Adam 算法中，梯度都无法正常下降，测试准确率和训练准确率都几乎为 10%，这与随机猜测的准确率一样，说明模型没有进行任何学习。导致这种情况可能的原因或许是，由于感知机参数很多，解空间比较病态，使用 SGD Momentum 算法和 Adam 算法比较容易陷入一些局部解中，而随机梯度下降法反而能够更好地在解空间中游走，使得准确率能够上升。

五、比较并讨论线性分类器、MLP 和 CNN 模型在 CIFAR-10 图像分类任务上的性能区别

类别	模型	测试准确率	达到最大准确率的训练轮数	一轮训练耗时
线性分类器	Softmax 分类器	31.47%	21	0.97s
MLP	三层感知机（隐藏层神经元数量：128、32）	45.55%	75	1.35s
	三层感知机（隐藏层神经元数量：512、32）	49.54%	69	1.36s
	四层感知机（隐藏层神经元数量：2048、64、64）	91.43%	79	1.61s
CNN	LeNet 去掉第 1 层 Pooling 层	77.00%	79	1.17s
	LeNet 去掉全部 2 层 Pooling 层	93.15%	77	1.15s

可以看到，线性分类器虽然耗时最少，但是 21 轮后就已经无法继续拟合，准确率最低。MLP 虽然能够达到较高的准确率，但是其耗时最长，并且参数量非常大，一旦降低其参数量其性能会快速下降。CNN 在三者中性能最好，其能够达到最高的正确率，同时训练速度仅次于线性分类器。