HW 02

李靖昕
2400013201

01

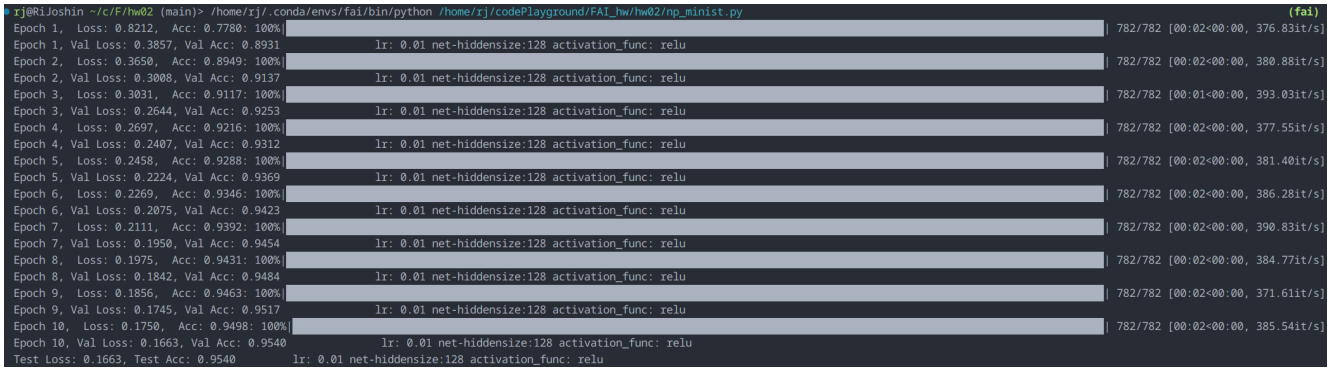
用numpy实现训练MLP网络识别手写数字MNIST数据集

改激活函数

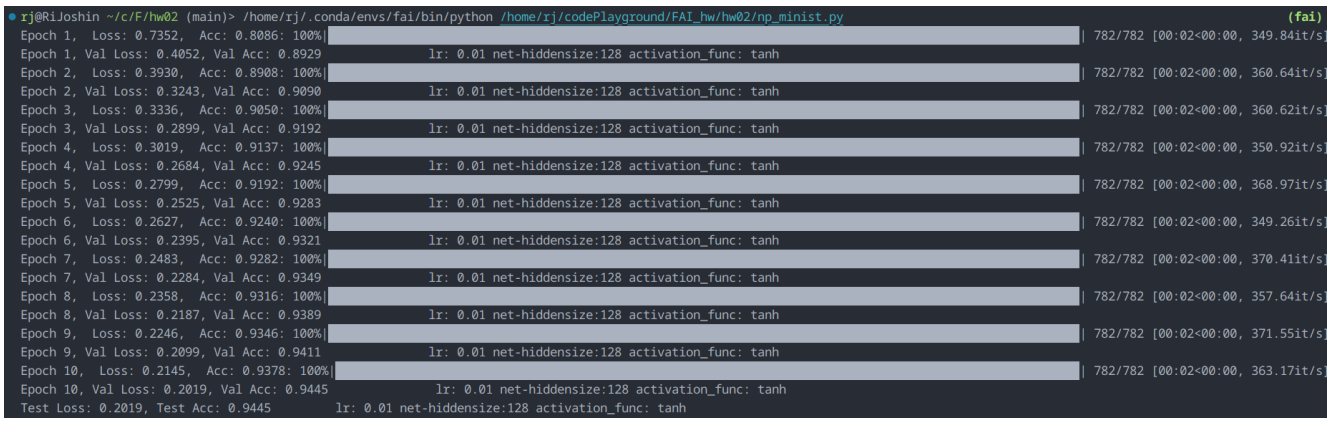
Net: 784 -> 128 -> 128 -> 10

Loss: CrossEntropy

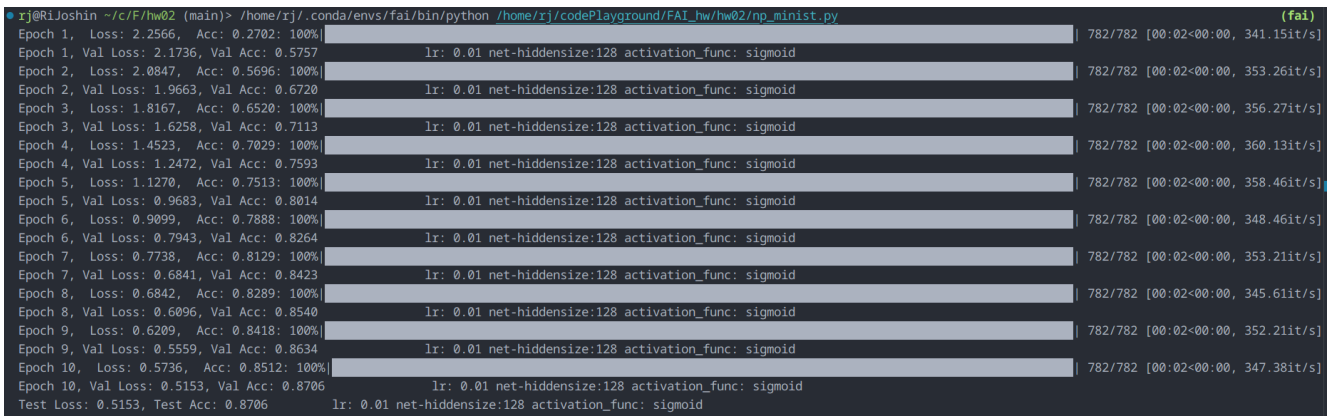
1. ReLU (Epoch 10, Test acc: 95.40%, Loss: 0.1663)



2. Tanh (Epoch 10, Test acc: 94.45%, Loss: 0.2019)



3. Sigmoid (Epoch 10, Test acc: 87.06%, Loss: 0.5153)



调整网络架构

- 784 → 800 → 800 → 10 (训练很慢，参数太多)

```
Epoch 1,  Loss: 0.5670,  Acc: 0.8480: 100%|| 782/782 [00:08<00:00, 89.32it/s]
Epoch 1, Val Loss: 0.3310, Val Acc: 0.9091          lr: 0.01 net-hiddensize:800 activation_func: tanh
Epoch 2,  Loss: 0.3353,  Acc: 0.9064: 100%|| 782/782 [00:09<00:00, 86.83it/s]
Epoch 2, Val Loss: 0.2849, Val Acc: 0.9206          lr: 0.01 net-hiddensize:800 activation_func: tanh
Epoch 3,  Loss: 0.2973,  Acc: 0.9169: 100%|| 782/782 [00:09<00:00, 79.37it/s]
Epoch 3, Val Loss: 0.2629, Val Acc: 0.9270          lr: 0.01 net-hiddensize:800 activation_func: tanh
Epoch 4,  Loss: 0.2746,  Acc: 0.9228: 100%|| 782/782 [00:11<00:00, 69.55it/s]
Epoch 4, Val Loss: 0.2475, Val Acc: 0.9301          lr: 0.01 net-hiddensize:800 activation_func: tanh
Epoch 5,  Loss: 0.2574,  Acc: 0.9273: 100%|| 782/782 [00:11<00:00, 68.82it/s]
Epoch 5, Val Loss: 0.2349, Val Acc: 0.9334          lr: 0.01 net-hiddensize:800 activation_func: tanh
Epoch 6,  Loss: 0.2429,  Acc: 0.9311: 100%|| 782/782 [00:10<00:00, 72.21it/s]
Epoch 6, Val Loss: 0.2239, Val Acc: 0.9383          lr: 0.01 net-hiddensize:800 activation_func: tanh
Epoch 7,  Loss: 0.2300,  Acc: 0.9350: 100%|| 782/782 [00:09<00:00, 86.16it/s]
Epoch 7, Val Loss: 0.2138, Val Acc: 0.9414          lr: 0.01 net-hiddensize:800 activation_func: tanh
Epoch 8,  Loss: 0.2181,  Acc: 0.9385: 100%|| 782/782 [00:09<00:00, 78.67it/s]
Epoch 8, Val Loss: 0.2045, Val Acc: 0.9448          lr: 0.01 net-hiddensize:800 activation_func: tanh
Epoch 9,  Loss: 0.2072,  Acc: 0.9417: 100%|| 782/782 [00:09<00:00, 84.57it/s]
Epoch 9, Val Loss: 0.1959, Val Acc: 0.9471          lr: 0.01 net-hiddensize:800 activation_func: tanh
Epoch 10,  Loss: 0.1970,  Acc: 0.9446: 100%|| 782/782 [00:09<00:00, 83.07it/s]
Epoch 10, Val Loss: 0.1879, Val Acc: 0.9492           lr: 0.01 net-hiddensize:800 activation_func: tanh
Test Loss: 0.1879, Test Acc: 0.9492         lr: 0.01 net-hiddensize:800 activation_func: tanh
```

- 784 → 200 → 200 → 10 （速度差为10X，相对上一个）

```
Epoch 1,  Loss: 0.6783,  Acc: 0.8234: 100%|| 782/782 [00:01<00:00, 416.33it/s]
Epoch 1, Val Loss: 0.3810, Val Acc: 0.8983          lr: 0.01 net-hiddensize:200 activation_func: tanh
Epoch 2,  Loss: 0.3768,  Acc: 0.8957: 100%|| 782/782 [00:01<00:00, 442.84it/s]
Epoch 2, Val Loss: 0.3142, Val Acc: 0.9145          lr: 0.01 net-hiddensize:200 activation_func: tanh
Epoch 3,  Loss: 0.3263,  Acc: 0.9074: 100%|| 782/782 [00:02<00:00, 390.43it/s]
Epoch 3, Val Loss: 0.2843, Val Acc: 0.9211          lr: 0.01 net-hiddensize:200 activation_func: tanh
Epoch 4,  Loss: 0.2980,  Acc: 0.9150: 100%|| 782/782 [00:02<00:00, 344.28it/s]
Epoch 4, Val Loss: 0.2647, Val Acc: 0.9263          lr: 0.01 net-hiddensize:200 activation_func: tanh
Epoch 5,  Loss: 0.2775,  Acc: 0.9210: 100%|| 782/782 [00:03<00:00, 201.75it/s]
Epoch 5, Val Loss: 0.2496, Val Acc: 0.9300          lr: 0.01 net-hiddensize:200 activation_func: tanh
Epoch 6,  Loss: 0.2609,  Acc: 0.9259: 100%|| 782/782 [00:03<00:00, 241.53it/s]
Epoch 6, Val Loss: 0.2369, Val Acc: 0.9338          lr: 0.01 net-hiddensize:200 activation_func: tanh
Epoch 7,  Loss: 0.2467,  Acc: 0.9298: 100%|| 782/782 [00:01<00:00, 402.83it/s]
Epoch 7, Val Loss: 0.2259, Val Acc: 0.9372          lr: 0.01 net-hiddensize:200 activation_func: tanh
Epoch 8,  Loss: 0.2339,  Acc: 0.9334: 100%|| 782/782 [00:02<00:00, 308.27it/s]
Epoch 8, Val Loss: 0.2159, Val Acc: 0.9398          lr: 0.01 net-hiddensize:200 activation_func: tanh
Epoch 9,  Loss: 0.2224,  Acc: 0.9367: 100%|| 782/782 [00:02<00:00, 300.54it/s]
Epoch 9, Val Loss: 0.2068, Val Acc: 0.9428          lr: 0.01 net-hiddensize:200 activation_func: tanh
Epoch 10,  Loss: 0.2119,  Acc: 0.9398: 100%|| 782/782 [00:02<00:00, 316.68it/s]
Epoch 10, Val Loss: 0.1986, Val Acc: 0.9462           lr: 0.01 net-hiddensize:200 activation_func: tanh
Test Loss: 0.1986, Test Acc: 0.9462         lr: 0.01 net-hiddensize:200 activation_func: tanh
```

- 784 → 20 → 10 （意料之中很低，中间特征太少，保存不了太多信息）

```
Epoch 1,  Loss: 1.9200,  Acc: 0.2679: 100%|| 782/782 [00:00<00:00, 5608.87it/s]
Epoch 1, Val Loss: 1.7234, Val Acc: 0.3579          lr: 0.01 net-hiddensize:2 activation_func: tanh
Epoch 2,  Loss: 1.6579,  Acc: 0.3655: 100%|| 782/782 [00:00<00:00, 5043.63it/s]
Epoch 2, Val Loss: 1.5752, Val Acc: 0.3943          lr: 0.01 net-hiddensize:2 activation_func: tanh
Epoch 3,  Loss: 1.5464,  Acc: 0.4241: 100%|| 782/782 [00:00<00:00, 5864.97it/s]
Epoch 3, Val Loss: 1.4898, Val Acc: 0.4520          lr: 0.01 net-hiddensize:2 activation_func: tanh
Epoch 4,  Loss: 1.4794,  Acc: 0.4631: 100%|| 782/782 [00:00<00:00, 5812.52it/s]
Epoch 4, Val Loss: 1.4354, Val Acc: 0.4808          lr: 0.01 net-hiddensize:2 activation_func: tanh
Epoch 5,  Loss: 1.4346,  Acc: 0.4843: 100%|| 782/782 [00:00<00:00, 5142.84it/s]
Epoch 5, Val Loss: 1.3968, Val Acc: 0.4934          lr: 0.01 net-hiddensize:2 activation_func: tanh
Epoch 6,  Loss: 1.4016,  Acc: 0.4935: 100%|| 782/782 [00:00<00:00, 5758.80it/s]
Epoch 6, Val Loss: 1.3670, Val Acc: 0.5014          lr: 0.01 net-hiddensize:2 activation_func: tanh
Epoch 7,  Loss: 1.3756,  Acc: 0.5029: 100%|| 782/782 [00:00<00:00, 5746.78it/s]
Epoch 7, Val Loss: 1.3424, Val Acc: 0.5105          lr: 0.01 net-hiddensize:2 activation_func: tanh
Epoch 8,  Loss: 1.3541,  Acc: 0.5148: 100%|| 782/782 [00:00<00:00, 5771.33it/s]
Epoch 8, Val Loss: 1.3213, Val Acc: 0.5293          lr: 0.01 net-hiddensize:2 activation_func: tanh
Epoch 9,  Loss: 1.3353,  Acc: 0.5343: 100%|| 782/782 [00:00<00:00, 4761.18it/s]
Epoch 9, Val Loss: 1.3025, Val Acc: 0.5548          lr: 0.01 net-hiddensize:2 activation_func: tanh
Epoch 10,  Loss: 1.3182,  Acc: 0.5577: 100%|| 782/782 [00:00<00:00, 5785.78it/s]
Epoch 10, Val Loss: 1.2854, Val Acc: 0.5832           lr: 0.01 net-hiddensize:2 activation_func: tanh
Test Loss: 1.2854, Test Acc: 0.5832         lr: 0.01 net-hiddensize:2 activation_func: tanh
```

- 784 → 128 → 10

```
Epoch 1,  Loss: 0.8579,  Acc: 0.7849: 100%|| 782/782 [00:01<00:00, 728.09it/s]
Epoch 1, Val Loss: 0.4714, Val Acc: 0.8866          lr: 0.01 net-hiddensize:128 activation_func: tanh
Epoch 2,  Loss: 0.4550,  Acc: 0.8810: 100%|| 782/782 [00:01<00:00, 745.08it/s]
Epoch 2, Val Loss: 0.3714, Val Acc: 0.9023          lr: 0.01 net-hiddensize:128 activation_func: tanh
Epoch 3,  Loss: 0.3869,  Acc: 0.8943: 100%|| 782/782 [00:01<00:00, 559.07it/s]
Epoch 3, Val Loss: 0.3328, Val Acc: 0.9076          lr: 0.01 net-hiddensize:128 activation_func: tanh
Epoch 4,  Loss: 0.3533,  Acc: 0.9018: 100%|| 782/782 [00:01<00:00, 526.97it/s]
Epoch 4, Val Loss: 0.3110, Val Acc: 0.9124          lr: 0.01 net-hiddensize:128 activation_func: tanh
Epoch 5,  Loss: 0.3318,  Acc: 0.9068: 100%|| 782/782 [00:02<00:00, 379.50it/s]
Epoch 5, Val Loss: 0.2960, Val Acc: 0.9160          lr: 0.01 net-hiddensize:128 activation_func: tanh
Epoch 6,  Loss: 0.3161,  Acc: 0.9111: 100%|| 782/782 [00:03<00:00, 247.65it/s]
Epoch 6, Val Loss: 0.2847, Val Acc: 0.9201          lr: 0.01 net-hiddensize:128 activation_func: tanh
Epoch 7,  Loss: 0.3035,  Acc: 0.9141: 100%|| 782/782 [00:04<00:00, 191.47it/s]
Epoch 7, Val Loss: 0.2754, Val Acc: 0.9223          lr: 0.01 net-hiddensize:128 activation_func: tanh
Epoch 8,  Loss: 0.2930,  Acc: 0.9172: 100%|| 782/782 [00:01<00:00, 673.30it/s]
Epoch 8, Val Loss: 0.2674, Val Acc: 0.9247          lr: 0.01 net-hiddensize:128 activation_func: tanh
Epoch 9,  Loss: 0.2838,  Acc: 0.9195: 100%|| 782/782 [00:01<00:00, 658.23it/s]
Epoch 9, Val Loss: 0.2604, Val Acc: 0.9268          lr: 0.01 net-hiddensize:128 activation_func: tanh
Epoch 10,  Loss: 0.2756,  Acc: 0.9218: 100%|| 782/782 [00:01<00:00, 503.44it/s]
Epoch 10, Val Loss: 0.2540, Val Acc: 0.9291           lr: 0.01 net-hiddensize:128 activation_func: tanh
Test Loss: 0.2540, Test Acc: 0.9291         lr: 0.01 net-hiddensize:128 activation_func: tanh
```

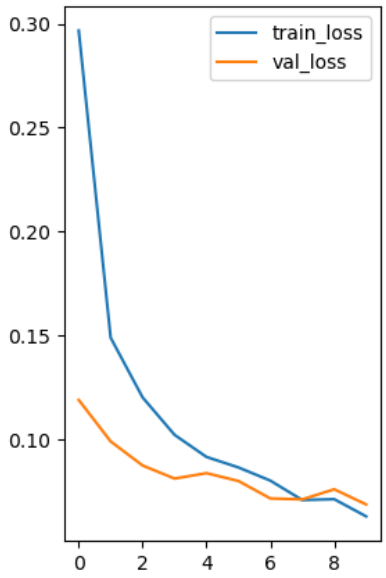
02

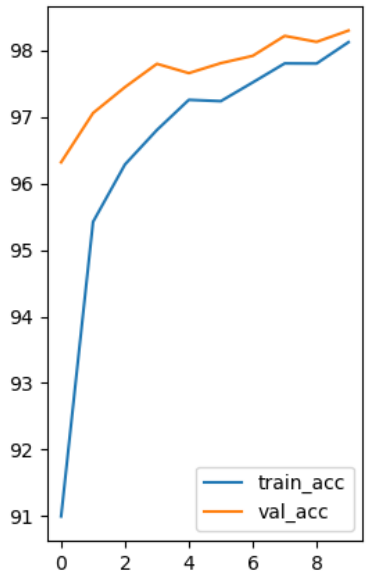
> 使用Pytorch训练MNIST数据集的MLP模型任务介绍

Adam + Dropout(0.5)

(Epoch: 10, Loss: 0.0687, Acc: 98.30%)

```
rj@RiJoshin ~/c/F/hw02 (main)> /home/rj/.conda/envs/fai/bin/python /home/rj/codePlayground/FAI_hw/hw02/mnist_mlp.py
Epoch 1:
Train set       : Average loss: 0.2967, Accuracy: 45494/50000 (90.99%)
Validation set  : Average loss: 0.1190, Accuracy:  9632/10000 (96.32%)

Epoch 2:
Train set       : Average loss: 0.1489, Accuracy: 47716/50000 (95.43%)
Validation set  : Average loss: 0.0991, Accuracy:  9706/10000 (97.06%)

Epoch 3:
Train set       : Average loss: 0.1202, Accuracy: 48145/50000 (96.29%)
Validation set  : Average loss: 0.0875, Accuracy:  9745/10000 (97.45%)

Epoch 4:
Train set       : Average loss: 0.1022, Accuracy: 48401/50000 (96.80%)
Validation set  : Average loss: 0.0812, Accuracy:  9780/10000 (97.80%)

Epoch 5:
Train set       : Average loss: 0.0916, Accuracy: 48631/50000 (97.26%)
Validation set  : Average loss: 0.0838, Accuracy:  9766/10000 (97.66%)

Epoch 6:
Train set       : Average loss: 0.0865, Accuracy: 48618/50000 (97.24%)
Validation set  : Average loss: 0.0800, Accuracy:  9781/10000 (97.81%)

Epoch 7:
Train set       : Average loss: 0.0802, Accuracy: 48762/50000 (97.52%)
Validation set  : Average loss: 0.0716, Accuracy:  9792/10000 (97.92%)

Epoch 8:
Train set       : Average loss: 0.0709, Accuracy: 48905/50000 (97.81%)
Validation set  : Average loss: 0.0712, Accuracy:  9822/10000 (98.22%)

Epoch 9:
Train set       : Average loss: 0.0713, Accuracy: 48904/50000 (97.81%)
Validation set  : Average loss: 0.0760, Accuracy:  9813/10000 (98.13%)

Epoch 10:
Train set       : Average loss: 0.0630, Accuracy: 49065/50000 (98.13%)
Validation set  : Average loss: 0.0687, Accuracy:  9830/10000 (98.30%)
```





Epoch	Train Acc	Val Acc
1	91.00	96.32
2	95.42	97.06
3	96.29	97.45
4	96.81	97.80
5	97.26	97.66
6	97.24	97.81
7	97.52	97.92
8	97.81	98.22
9	97.80	98.13
10	98.13	98.30

部分网络结构试验	Train_Acc(%)	Val_Acc(%)
784-Dropout(.5)-800-Dropout(.5)-800-10 + Adam	98.12	98.30
784-Dropout(.5)-1000-Dropout(.5)-800-10 + Adam	99.40	98.16
784-Dropout(.1)-1000-Dropout(.5)-800-10 + Adam	99.49	97.72
784-Dropout(.9)-1000-Dropout(.5)-800-10 + Adam	99.48	97.80
784-1000-800-10 + Adam	99.35	97.79
784-800-800-10 + Adam	99.44	98.06
784-800-800-10 + SGD	82.05	84.66
784-Dropout(.5)-900-Dropout(.5)-800-10-RMSprop-	97.52	97.44
784-900-800-10-RMSprop	99.51	97.54