



南開大學  
Nankai University

南 開 大 學

計 算 機 學 院

深度學習與應用實驗報告

---

## MLP 前饋神經網絡

---

艾明旭 2111033

年 級：2021 級

專 業：信息安全

指導教師：侯淇彬

2024 年 3 月 18 日

## 目录

<b>一、 基础 MLP 结构</b>	<b>1</b>
(一) 基础 MLP 的网络结构 . . . . .	1
(二) 图形展示实验结果 . . . . .	1
(三) 增大迭代次数 . . . . .	1
(四) 增大网络层数 . . . . .	1
<b>二、 ResNet</b>	<b>3</b>
(一) ResNet 的实现 . . . . .	3
(二) 结果图像展示 . . . . .	4
<b>三、 总结</b>	<b>4</b>

## 一、基础 MLP 结构

### (一) 基础 MLP 的网络结构

```
1 Net(  
2   fc1: Linear(784, 100, float32[100,], None)  
3   fc1_drop: Dropout(0.2, is_train=False)  
4   fc2: Linear(100, 80, float32[80,], None)  
5   fc2_drop: Dropout(0.2, is_train=False)  
6   fc3: Linear(80, 10, float32[10,], None)  
7 )
```

基础的 MLP 网络为一个三层的网络，其映射关系如上面的结构所示。可以通过 python 代码进行相应的输出。

### (二) 图形展示实验结果

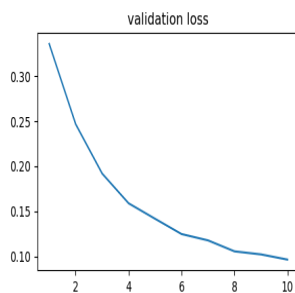


图 1: 基础 MLP validation loss

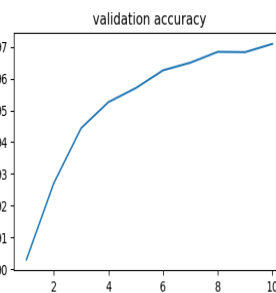


图 2: 基础 MLP validation accuracy

### (三) 增大迭代次数

接下来我选择将迭代次数增加，得到了结果更加平滑的曲线。说明网络取得了一个更好的结果。

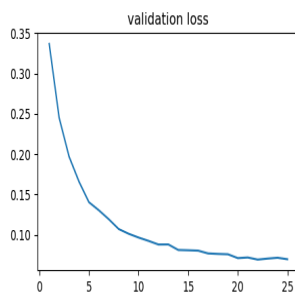


图 3: 加深 validation loss

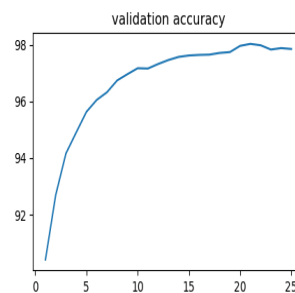


图 4: 加深 validation accuracy

### (四) 增大网络层数

我尝试将基础 MLP 的网络层数加深，得到了更加平滑的曲线，结果更加完善。

**首先是四层神经网络** 可以发现曲线有一定的改变。

```

1 self.fc1 = nn.Linear(28*28, 512) # weight: [28*28, 50]   bias: [50, ]
2 self.fc1_drop = nn.Dropout(0.2)
3 self.fc2 = nn.Linear(512, 256)
4 self.fc2_drop = nn.Dropout(0.2)
5 self.fc3 = nn.Linear(256, 128)
6 self.fc3_drop = nn.Dropout(0.2)
7 self.fc4 = nn.Linear(128, 10)

```

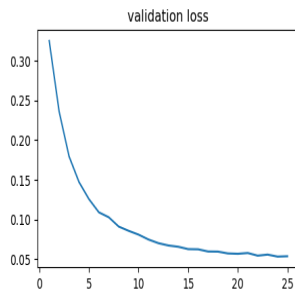


图 5: 4 层 validation loss

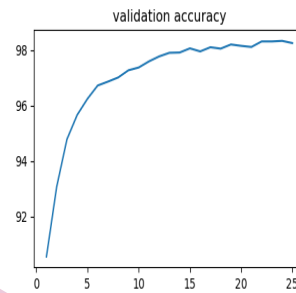


图 6: 4 层 validation accuracy

**接下来尝试六层神经网络** 可以看到曲线拟合程度的提升。

```

1 self.fc1 = nn.Linear(28*28, 512) # weight: [28*28, 50]   bias: [50, ]
2 self.fc1_drop = nn.Dropout(0.2)
3 self.fc2 = nn.Linear(512, 256)
4 self.fc2_drop = nn.Dropout(0.2)
5 self.fc3 = nn.Linear(256, 128)
6 self.fc3_drop = nn.Dropout(0.2)
7 self.fc4 = nn.Linear(128, 64)
8 self.fc4_drop = nn.Dropout(0.2)
9 self.fc5 = nn.Linear(64, 32)
10 self.fc5_drop = nn.Dropout(0.2)
11 self.fc6 = nn.Linear(32, 10)

```

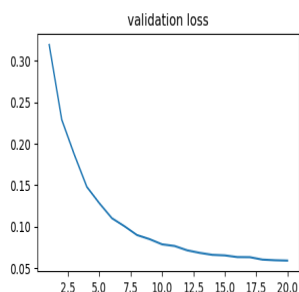


图 7: 6 层 validation loss

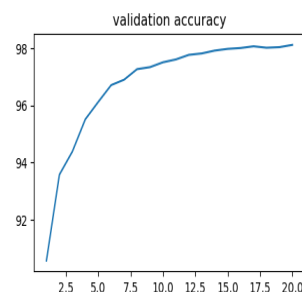


图 8: 6 层 validation accuracy

## 二、 ResNet

### (一) ResNet 的实现

ResNet (Residual Neural Network) 是一种深度神经网络结构, 由微软研究院的研究员提出, 旨在解决深度神经网络训练过程中的梯度消失和梯度爆炸等问题。ResNet 通过引入残差模块 (Residual Block) 来构建深层网络, 使得网络可以更深更容易训练。

在 ResNet 中, 残差模块通过增加跨层连接 (skip connection) 来实现残差学习。传统的神经网络会将输入信号传递到激活函数, 而在 ResNet 中, 跨层连接使得输入信号可以直接绕过若干层的非线性变换, 从而保留输入信号的信息, 这样有助于减轻梯度消失的问题, 使得网络更容易训练。

```
1 class ResMLP(nn.Module):
2     def __init__(self, in_dim, hidden_dim, out_dim, num_layers):
3         super(ResMLP, self).__init__()
4         self.in_layer = nn.Linear(in_dim, hidden_dim)
5         self.hidden_layers = nn.ModuleList([nn.Linear(hidden_dim, hidden_dim)
6                                             for i in range(num_layers)])
7         self.out_layer = nn.Linear(hidden_dim, out_dim)
8         self.relu = nn.ReLU()
9
10    def forward(self, x):
11        x = self.in_layer(x)
12        for layer in self.hidden_layers:
13            x = x + self.relu(layer(x))
14        x = self.out_layer(x)
15        return x
```

定义参数过后, 输出网络格式。

```
1 ResMLP(
2   (in_layer): Linear(in_features=784, out_features=256, bias=True)
3   (hidden_layers): ModuleList(
4     (0): Sequential(
5       (0): Linear(in_features=256, out_features=256, bias=True)
6       (1): ReLU()
7       (2): Dropout(p=0.2, inplace=False)
8     )
9     (1): Sequential(
10      (0): Linear(in_features=256, out_features=256, bias=True)
11      (1): ReLU()
12      (2): Dropout(p=0.2, inplace=False)
13    )
14    (2): Sequential(
15      (0): Linear(in_features=256, out_features=256, bias=True)
16      (1): ReLU()
17      (2): Dropout(p=0.2, inplace=False)
18    )
19    (3): Sequential(
20      (0): Linear(in_features=256, out_features=256, bias=True)
```

```
21     (1): ReLU()
22     (2): Dropout(p=0.2, inplace=False)
23 )
24 (4): Sequential(
25     (0): Linear(in_features=256, out_features=256, bias=True)
26 ...
27 )
28 )
29 (out_layer): Linear(in_features=256, out_features=10, bias=True)
30 )
```

## (二) 结果图像展示

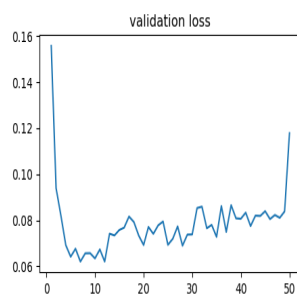


图 9: ResNet validation loss

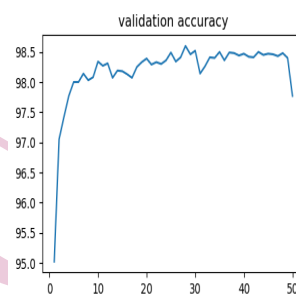


图 10: ResNet validation accuracy

可以看到，在引入残差块之后，整个深度学习的网络层次都发生了一定的变化。

## 三、 总结

我们可以看到随着学习率的升高，他的拟合速度变快，在训练 10 轮的情况下准确率会提升，但是同时的，其变化也更为陡峭，此外我们可以看到学习率在 0.05 的时候已经可以看到反而有训练多一轮反而准确率下降，同时 loss 上升，这代表着过大的学习率也会让学习过程中错过拟合函数的极值点，导致无法得到最优效果，所以在训练过程中，往往开始设置较大的学习率，在后面慢慢降低。

关于网络结构的调整，我将前面的维度提高，以期望可以在一开始可以多学习图片的特征，此外增加了一层额外的全连接层来增加网络的表达能力，实践表明，控制其他参数不变的情况下，确实比原始的网络效果有所提升。

对于 ResNet 的网络结构，我们尝试了引入残差块来实现网络层次的变化，最终我们发现数据集的图像结果存在一定的过拟合现象，需要我们对模型的参数等进行一定的调整。

本次实验当中，我对前馈神经网络有了更加深刻的理解，也能够编写出相应的代码，希望可以在未来的学下当中继续进步，在深度学习这门课当中学习到更多有用而充满挑战性的知识。