

深度学习与神经网络第二次课程项目

王逸群 19307110397

2022.4.9

1 神经网络

1.1 初始架构

本项目使用 CIFAR-10 数据集，其中包含 60000 张 32×32 的彩色图片，被平均分为 10 类：飞机、汽车、鸟、猫、鹿、狗、青蛙、马、船、货车。

参考 VGG 网络架构，基于 pytorch 框架，设计神经网络初始架构存储于 `Code/nn.py`，内容如下：

```
1 class NN(nn.Module):
2     def __init__(self, in_channels = 3, hidden_channels = (16, 32),
3                 hidden_neurons = (128, 128), num_classes = 10):
4         super().__init__()
5         self.hidden_channels = hidden_channels
6
7         self.extractor = nn.Sequential(
8             # stage 1
9             nn.Conv2d(in_channels = in_channels,
10                      out_channels = hidden_channels[0],
11                      kernel_size = 3, padding = 1),
12             nn.ReLU(),
13             nn.MaxPool2d(kernel_size = 2, stride = 2),
14
15             # stage 2
16             nn.Conv2d(in_channels = hidden_channels[0],
17                      out_channels = hidden_channels[1],
18                      kernel_size = 3, padding = 1),
19             nn.ReLU(),
20             nn.MaxPool2d(kernel_size = 2, stride = 2))
21
22         self.classifier = nn.Sequential(
23             nn.Linear(hidden_channels[1] * 8 * 8, hidden_neurons[0]),
```

```

24         nn.ReLU(),
25         nn.Linear(hidden_neurons[0], hidden_neurons[1]),
26         nn.ReLU(),
27         nn.Linear(hidden_neurons[1], num_classes))
28
29     def forward(self, inputs):
30         hidden = self.extractor(inputs)
31         outputs = \
32             self.classifier(hidden.view(-1,
33                                         self.hidden_channels[1] * 8 * 8))
34     return outputs

```

1.2 参数调整

1.2.1 神经元数量

1.2.2 损失函数

1.2.3 正则化

1.2.4 激活函数

1.2.5 优化器

1.2.6 批归一化

1.2.7 丢弃法

1.2.8 残差连接

1.3 可视化

1.3.1 卷积核可视化

1.3.2 损失函数曲线

2 批归一化