

第三次习题课

TA 孙婧雯

2025-03-26



内容一览

- 1 复习：损失函数与优化器
- 2 实操：from FNN to DNN
- 3 终端：更舒适地coding



优化器Optimizer

为什么需要优化器?

反向传播过程中，指引损失函数（目标函数）的各个参数往正确的方向更新合适的大小，使得更新后的各个参数让损失函数（目标函数）值不断逼近全局最小。

```
# 使用均方误差 (MSE) 作为损失函数
self.criterion = nn.MSELoss()

# 使用 SGD 优化器
self.optimizer = optim.SGD(self.model.parameters(), lr=lr)
```

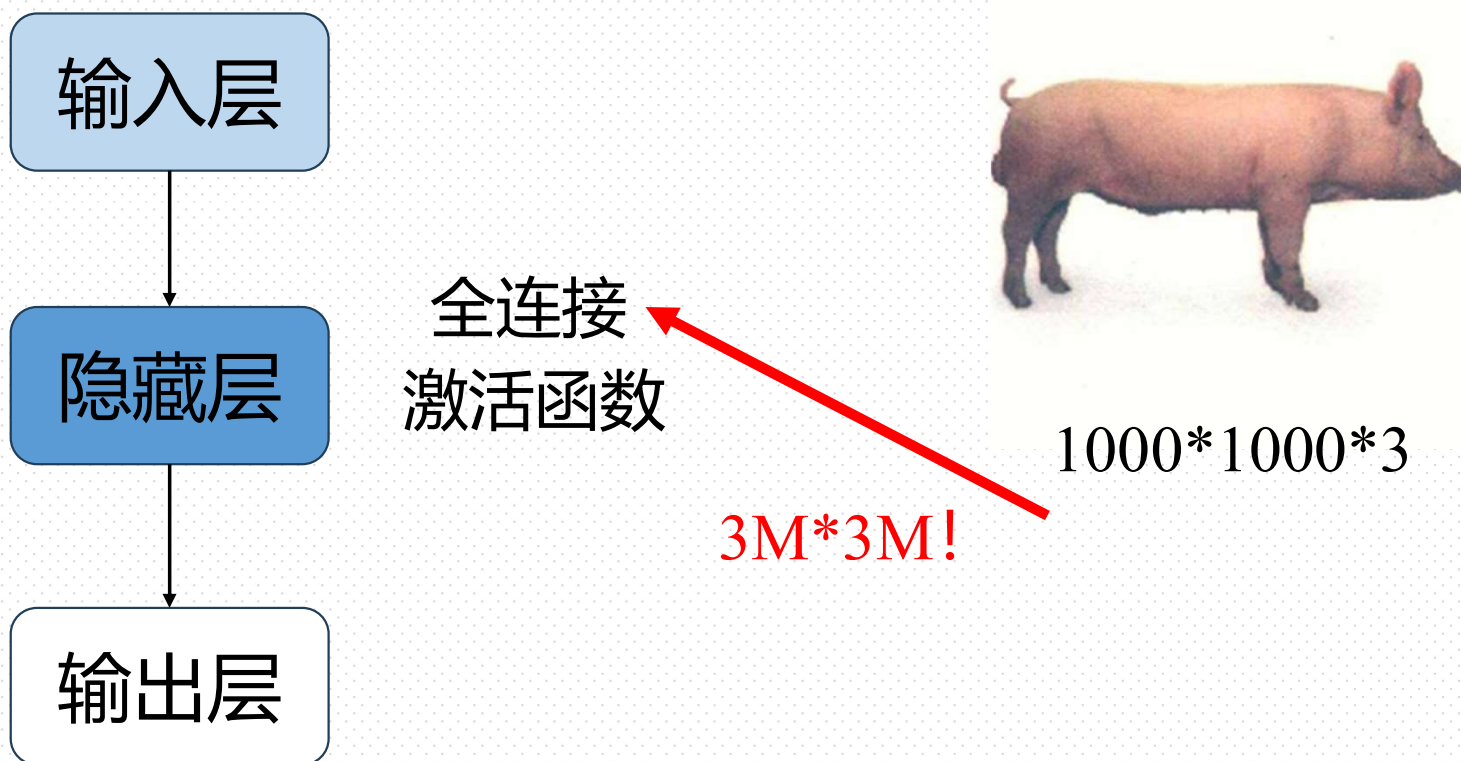


优化器Optimizer

- SGD 随机梯度下降
- SGDM 动量随机梯度下降
- AdaGrad 自适应梯度
- RMSProp “根号均方”
- **Adam 自适应动量估计**

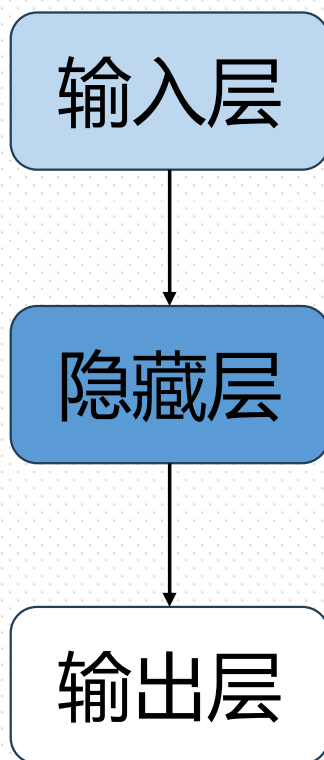


from FNN to DNN

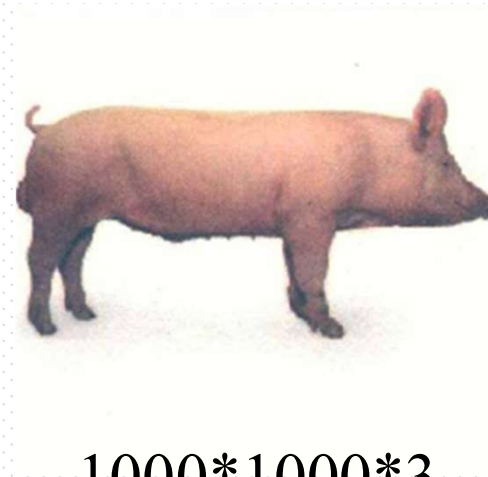
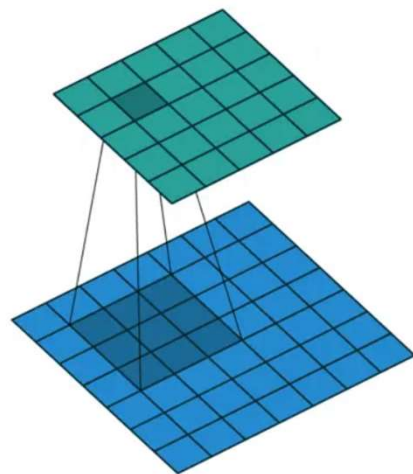




from FNN to DNN



卷积Convolution



$1000 \times 1000 \times 3$



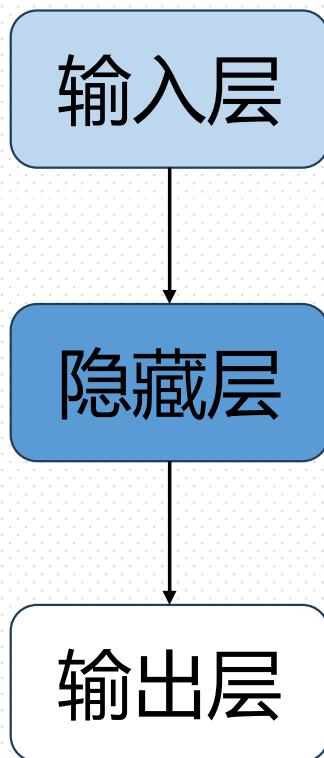
10×10 kernel

$3M \times 300$

Much less!



from FNN to DNN



卷积Convolution
池化Pooling

聚合统计特征
10*10 pooling



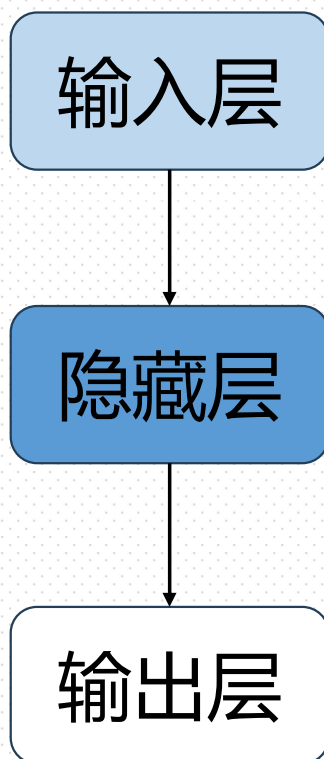
$1000*1000*3$
(hidden layer)



$100*100*3$
(hidden layer)



from FNN to DNN



示例：如果使用 2×2 最大池化：

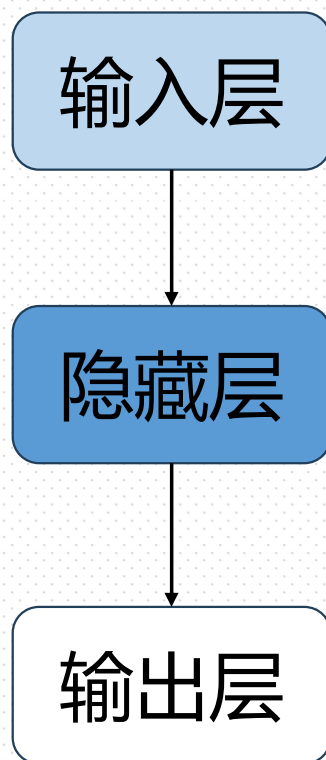
$$\begin{bmatrix} 1 & 3 & 2 & 4 \\ 5 & 6 & 7 & 8 \\ 3 & 2 & 1 & 0 \\ 9 & 8 & 7 & 6 \end{bmatrix}$$

经过 2×2 最大池化（步长2），得到：

$$\begin{bmatrix} 6 & 8 \\ 9 & 7 \end{bmatrix}$$

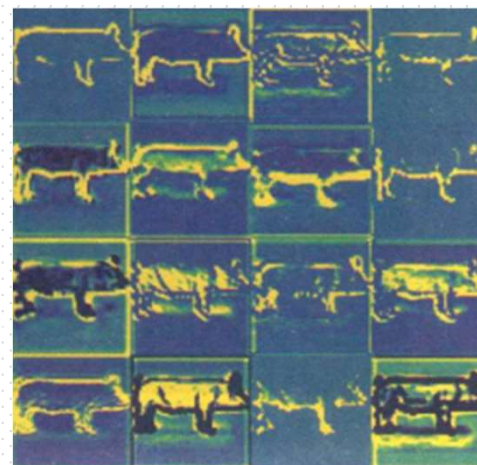
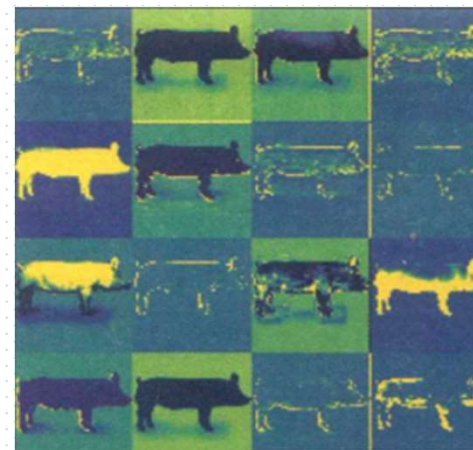


from FNN to DNN



卷积Convolution
池化Pooling

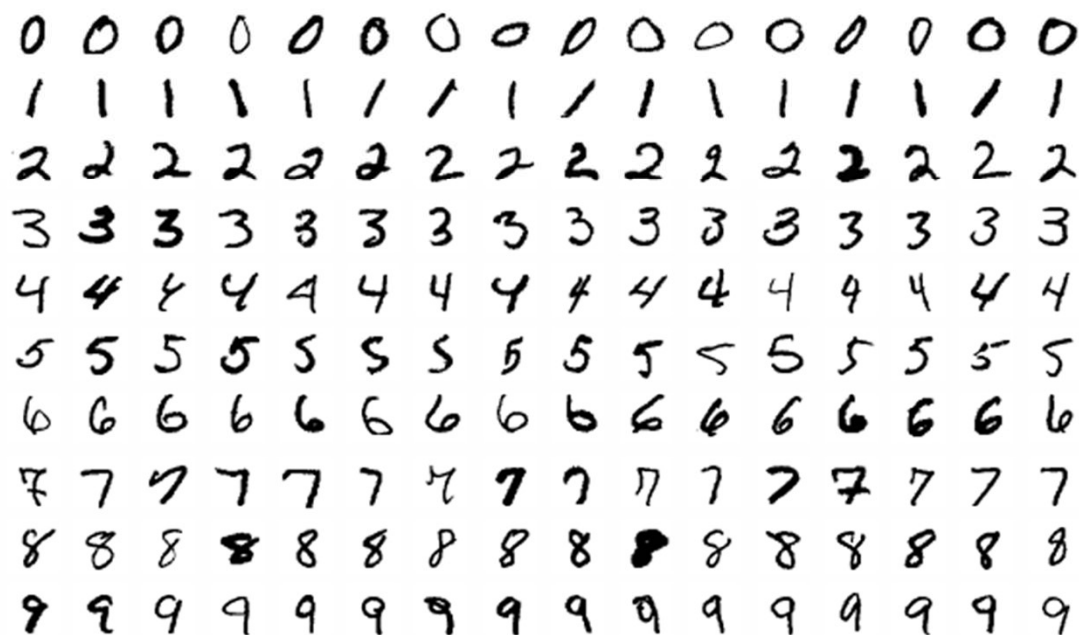
提取特征，参数共享
降采样，再激活连接





实操：手写数字识别

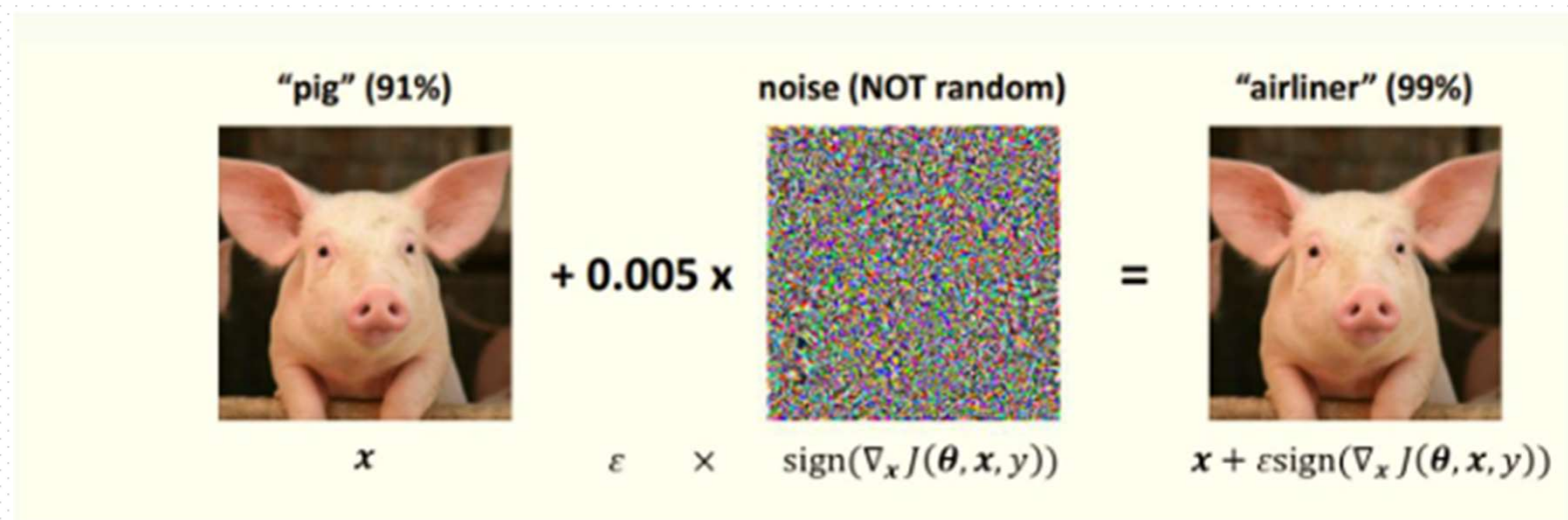
MNIST是一个手写数字数据库，常用于训练各种图像处理系统。它由来自250个不同人的手写数字组成，共有10类，对应于从0到9的手写数字。每个图像都是28x28像素的灰度图。其中60,000个示例通常用于训练，另外10,000个示例用于测试。





深度学习科研

- 图像处理识别
- 计算机视觉
- 人工智能安全
- 优化算法
-



Simple Approach to Fool Deep Neural Networks: Fast Gradient Sign Method(FSGM). [Goodfellow, et al. 2014]

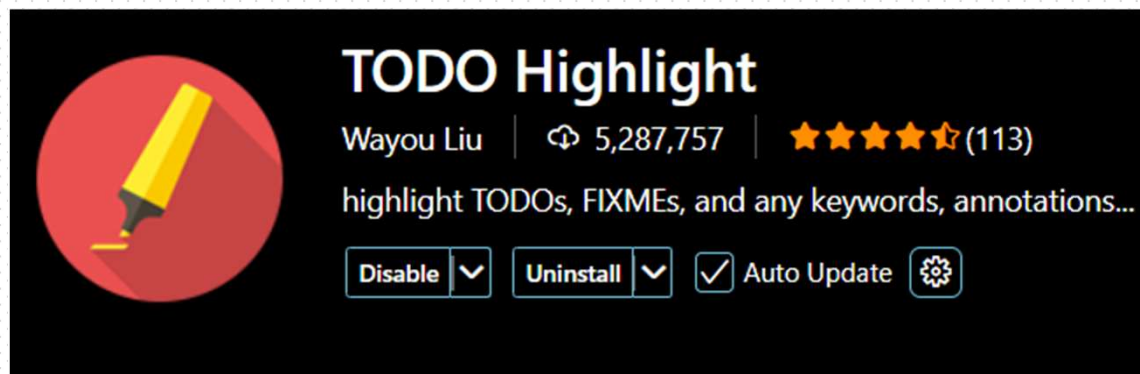


终端Terminal

- 一些快捷键: Ctrl + ` Ctrl + Shift + ` Ctrl + C 方向键↑ 方向键↓
- 新建终端
- 终端分屏
- 一些命令: cd mkdir ...
- Tab 自动补全

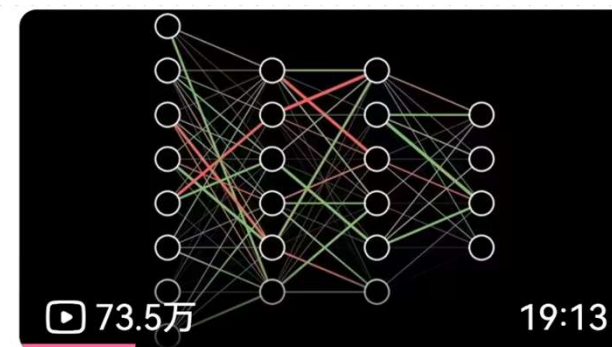


推荐



File > Preferences > Settings > Extensions

```
{
  "text": "TODO",
  "color": "#000",
  "isWholeLine": false,
},
{
  "text": "BEGIN_YOUR_CODE",
  "color": "#000",
  "isWholeLine": false,
},
```



【官方双语】深度学习之神经网络的结构 Part 1 ver 2.0

[【3Blue1Brown】深度学习之神经网络的结构 Part 1 ver 2.0](#)

Thank you

如有更多问题请下课/线上提出

TA 孙婧雯