



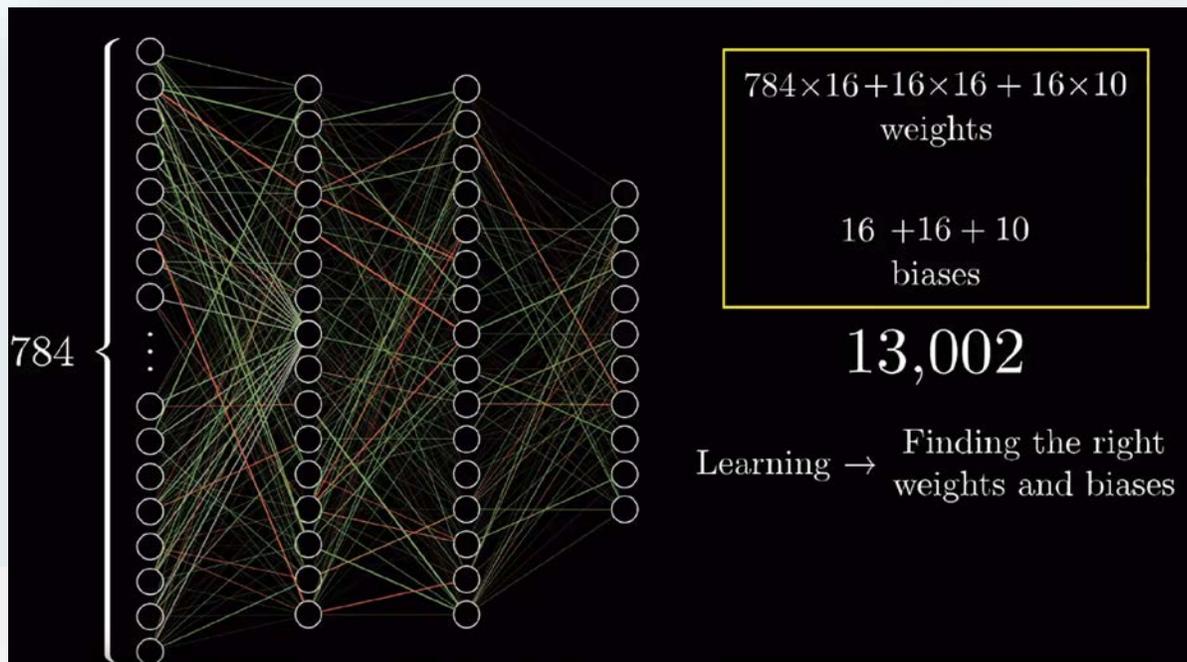
# 人工智能与信息社会

基于神经网络的智能系统I：计算损失函数

陈斌 北京大学 [gischen@pku.edu.cn](mailto:gischen@pku.edu.cn)

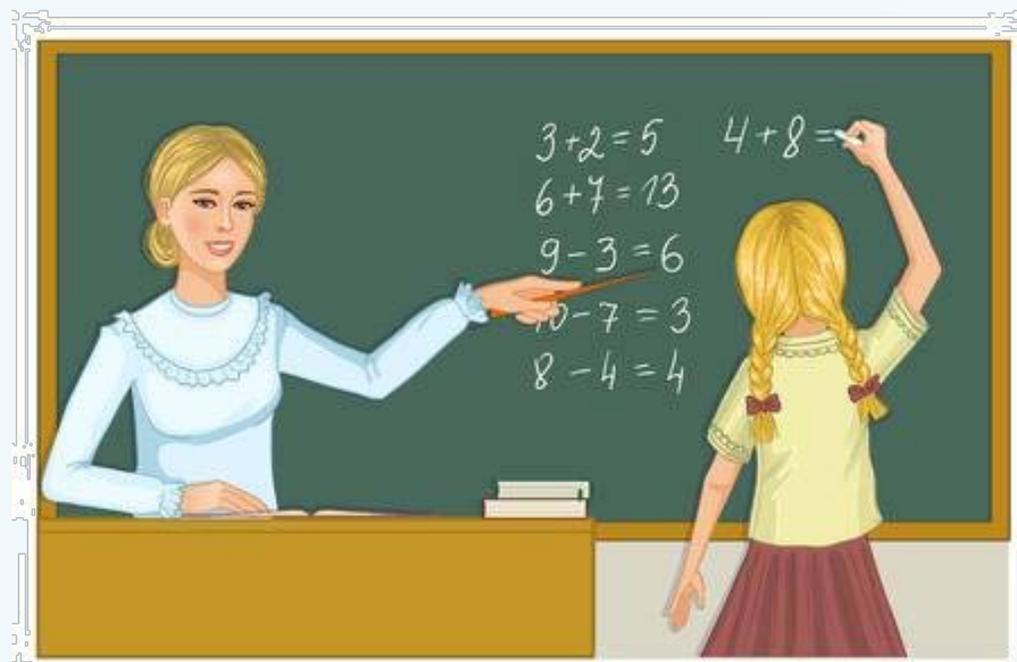
# 人工神经网络的训练

- › 在介绍了神经网络的主要模型并搭建完一个神经网络之后，我们就要对它进行训练。
- › 对于一个2层的小网络，就可能有一万三千多个参数需要我們进行计算。



# 人工神经网络的训练

- 我们对于网络的训练是通过一系列带有标记的样本数据来进行的，这种学习方式称为监督学习。



# 人工神经网络的训练

- › 既然样本数据带有标签，我们就要把神经网络预测得到的结果与标签进行对比，从而判断预测的准确性。
- › 如果发现预测结果不够好，那么我们将调整网络的参数，使得网络能够做出更好的预测。
- › 如何评价预测结果好坏？通过损失函数来定量计算。



# 损失函数

- › **损失函数 (loss function) 又叫做代价函数 (cost function), 是用来评估模型的预测值与真实值的误差大小。**
- › **神经网络训练的过程就是最小化损失函数的过程, 通过不断优化网络参数, 使模型的预测值更加接近真实值。**

# 损失函数

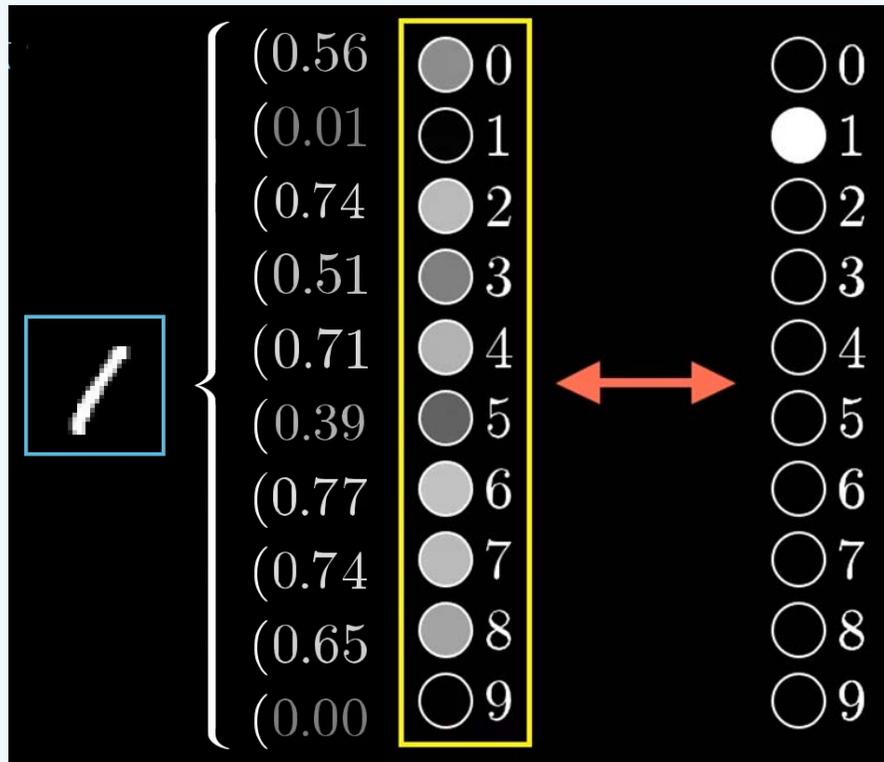
› **0-1损失函数：一种常用于分类任务的损失函数。**

预测值与真实值不同，就是预测错误，则损失是1；  
预测值与真实值相等，就是预测正确，损失是 0，  
就是没有损失。

$$l_i = \begin{cases} 0 & \text{如果}\vec{x}\text{不属于第}i\text{类} \\ 1 & \text{如果}\vec{x}\text{属于第}i\text{类} \end{cases}$$

# 损失函数

- › 在本实例中，不同的数字有不同的概率，不是非0即1，不适用于0-1损失函数



# 损失函数

- › 平方损失函数
- › 预测值与真实值的差的平方和。
- › 平方损失函数比0-1损失函数能够反映多的信息，在手写数字识别实例中更有利于参数调整。

$$L = (Y - f(x))^2$$

Y: 真实值（标签）

F(x): 预测值

# 损失函数

- 通过不断减小平方损失函数来提高分类的准确率。


$$\left\{ \begin{array}{l} (0.56 - 0.00)^2 + \\ (0.01 - 1.00)^2 + \\ (0.74 - 0.00)^2 + \\ (0.51 - 0.00)^2 + \\ (0.71 - 0.00)^2 + \\ (0.39 - 0.00)^2 + \\ (0.77 - 0.00)^2 + \\ (0.74 - 0.00)^2 + \\ (0.65 - 0.00)^2 + \\ (0.00 - 0.00)^2 \end{array} \right.$$

# 总结

- › 通过损失函数来定量计算神经网络预测结果的好坏。
- › 神经网络训练的过程就是最小化损失函数的过程
- › 损失函数越小说明预测结果越接近真实结果(标签)