AI MOOC

2

MNIST数据集介绍

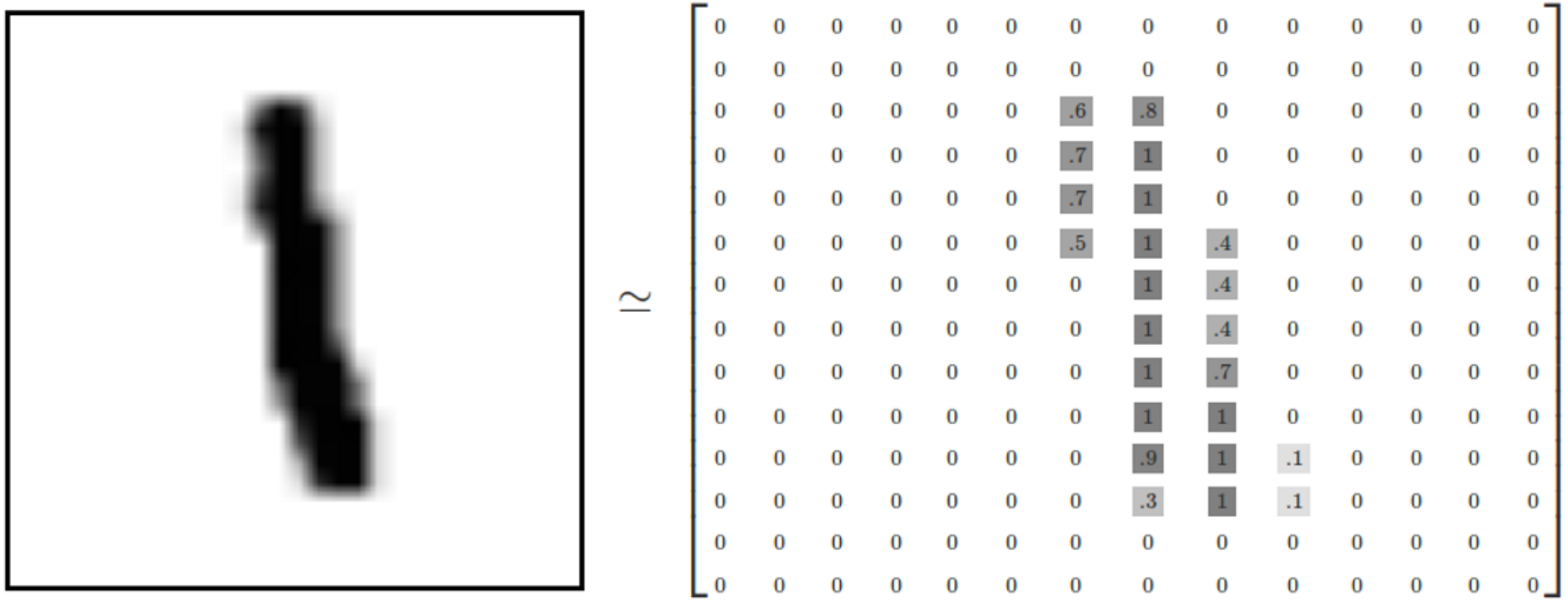
MNIST数据集

MNIST数据集官网：Yann LeCun’ s Website

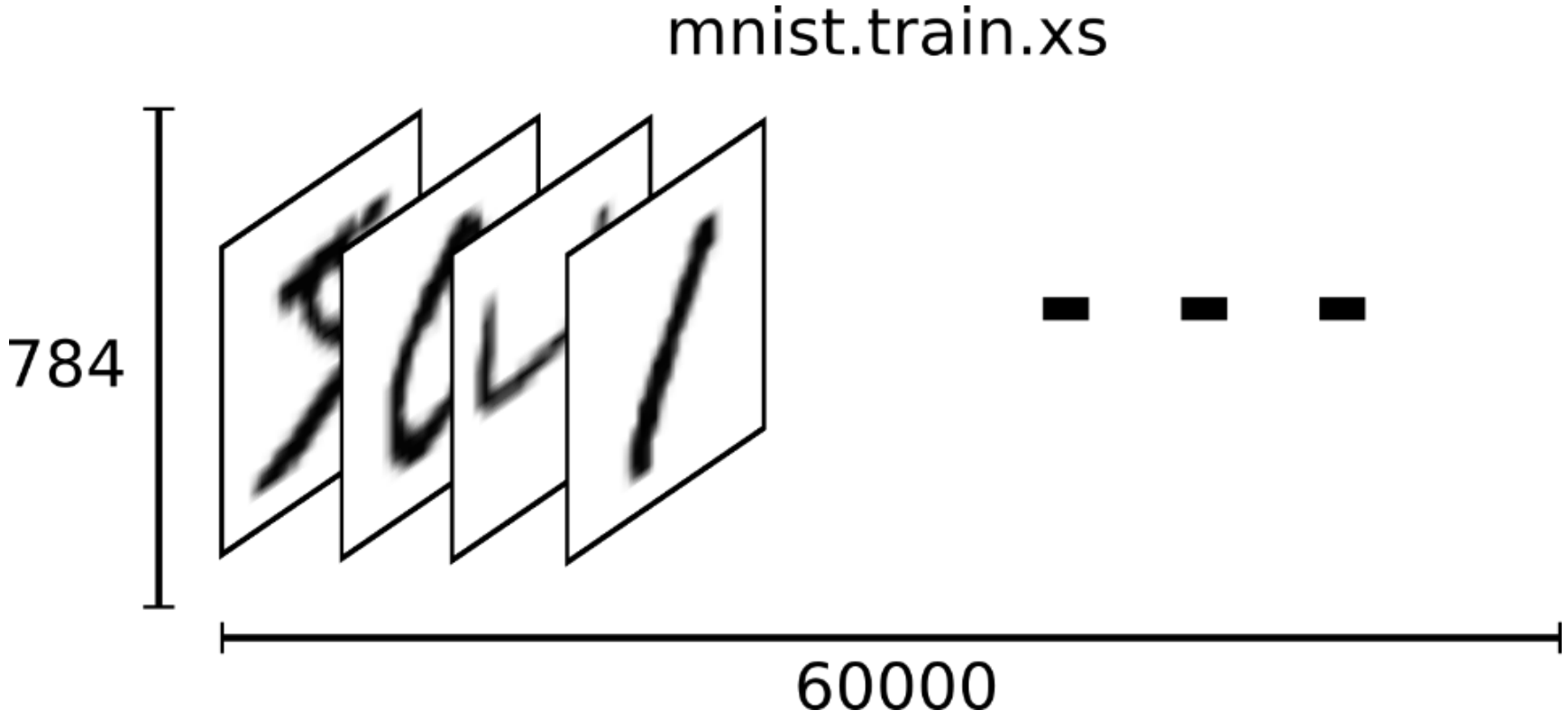
下载下来的数据集被分成两部分：60000行的训练数据集（mnist.train）和10000行的测试数据集（mnist.test）

MNIST的数据

一张图片包含28*28个像素，我们把这一个数组展开成一个向量，长度是28*28=784。如果把数据用矩阵表示，可以把MNIST训练数据变成一个形状为 [60000, 784] 的矩阵，第一个维度数字用来索引图片，第二个维度数字用来索引每张图片中的像素点。图片里的某个像素的强度值介于0-1之间。



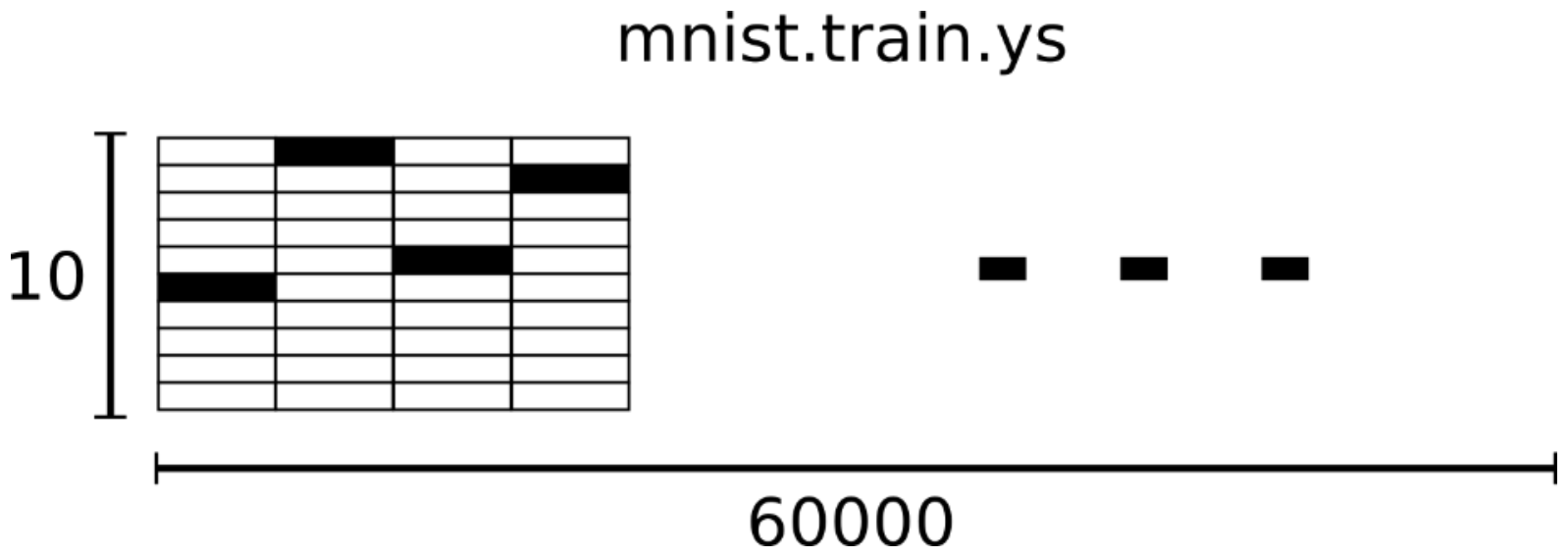
MNIST的数据



独热编码（one-hot）

- MNIST数据集的标签是介于0-9的数字，我们要把标签转化为“one-hot vectors”。一个one-hot向量除了某一位数字是1以外，其余维度数字都是0，比如标签0将表示为([1,0,0,0,0,0,0,0,0,0])，标签3将表示为([0,0,0,1,0,0,0,0,0,0])。

- 因此，可以把MNIST训练集的标签变为 [60000, 10] 的矩阵。



AI MOOC

3

Softmax函数介绍

Softmax激活函数

在多分类问题中，我们通常会使用softmax函数作为网络输出层的激活函数，softmax函数可以对输出值进行归一化操作，把所有输出值都转化为概率，所有概率值加起来等于1，softmax的公式为：

$$softmax(x)_i = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$

Softmax计算例子

例如某个神经网络有3个输出值，为[1,5,3]。

计算e1=2.718，e5=148.413，e3=20.086，e1+e5+e3=171.217。

p1= $\frac{e^1}{e^1+e^5+e^3} = 0.016$，p2 = $\frac{e^5}{e^1+e^5+e^3} = 0.867$，p3 = $\frac{e^3}{e^1+e^5+e^3} = 0.117$。

所以加上softmax函数后数值变成了[0.016,0.867,0.117]。

例如手写数字识别的网络最后的输出结果本来是:
[-0.124, -4.083, -0.62, 0.899, -1.193, -0.701, -2.834, 6.925, -0.332, 2.064]，
加上softmax函数后会变成:
[0.001, 0.0, 0.001, 0.002, 0.0, 0.0, 0.0, 0.987, 0.001, 0.008]。

交叉熵

二次代价函数

二次代价函数：

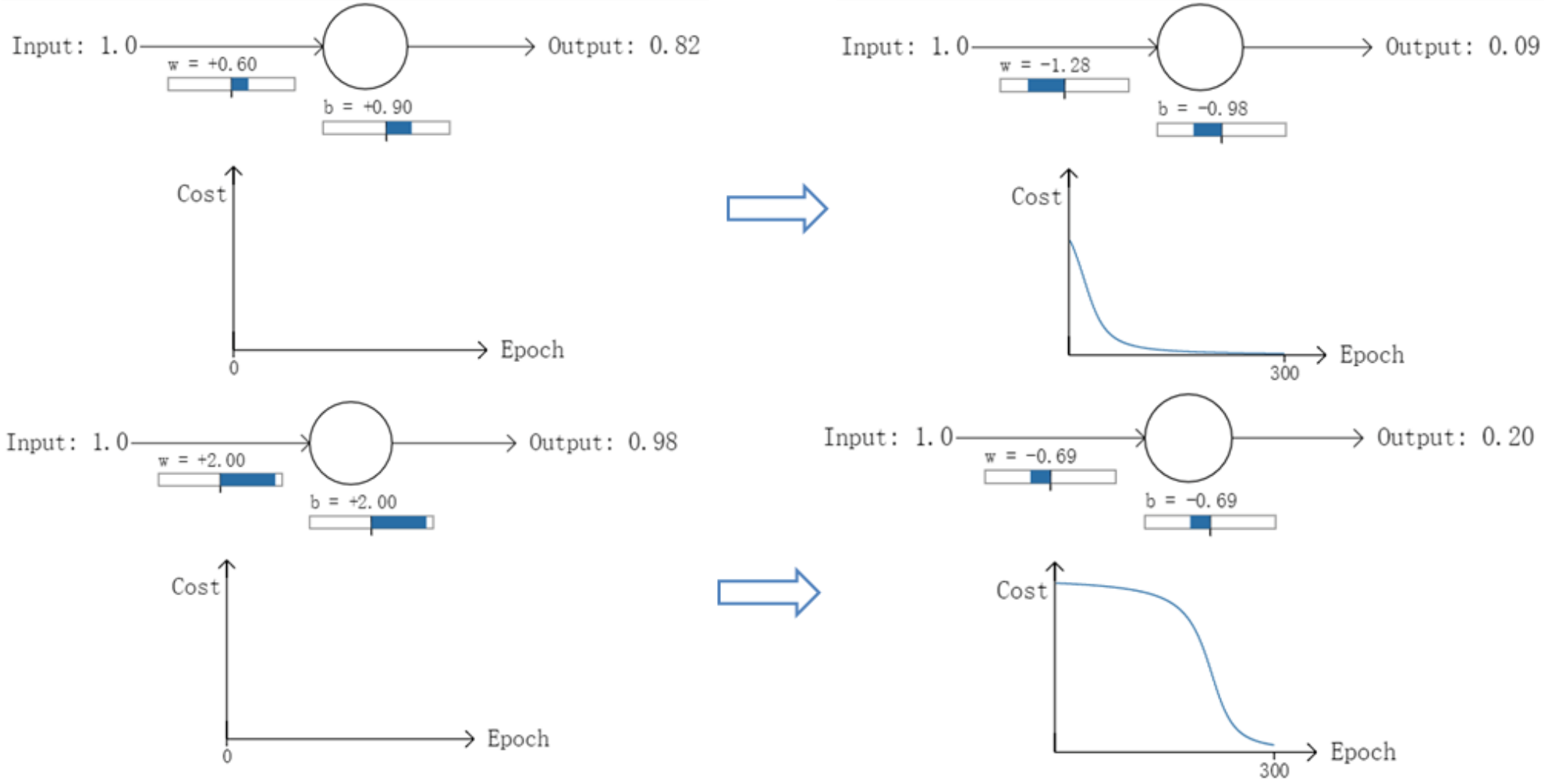
$$E = \frac{1}{2}(t-y)^2$$

$$\frac{\partial E}{\partial w} = (y-t)\,f'(z)\,x \qquad z = WX$$

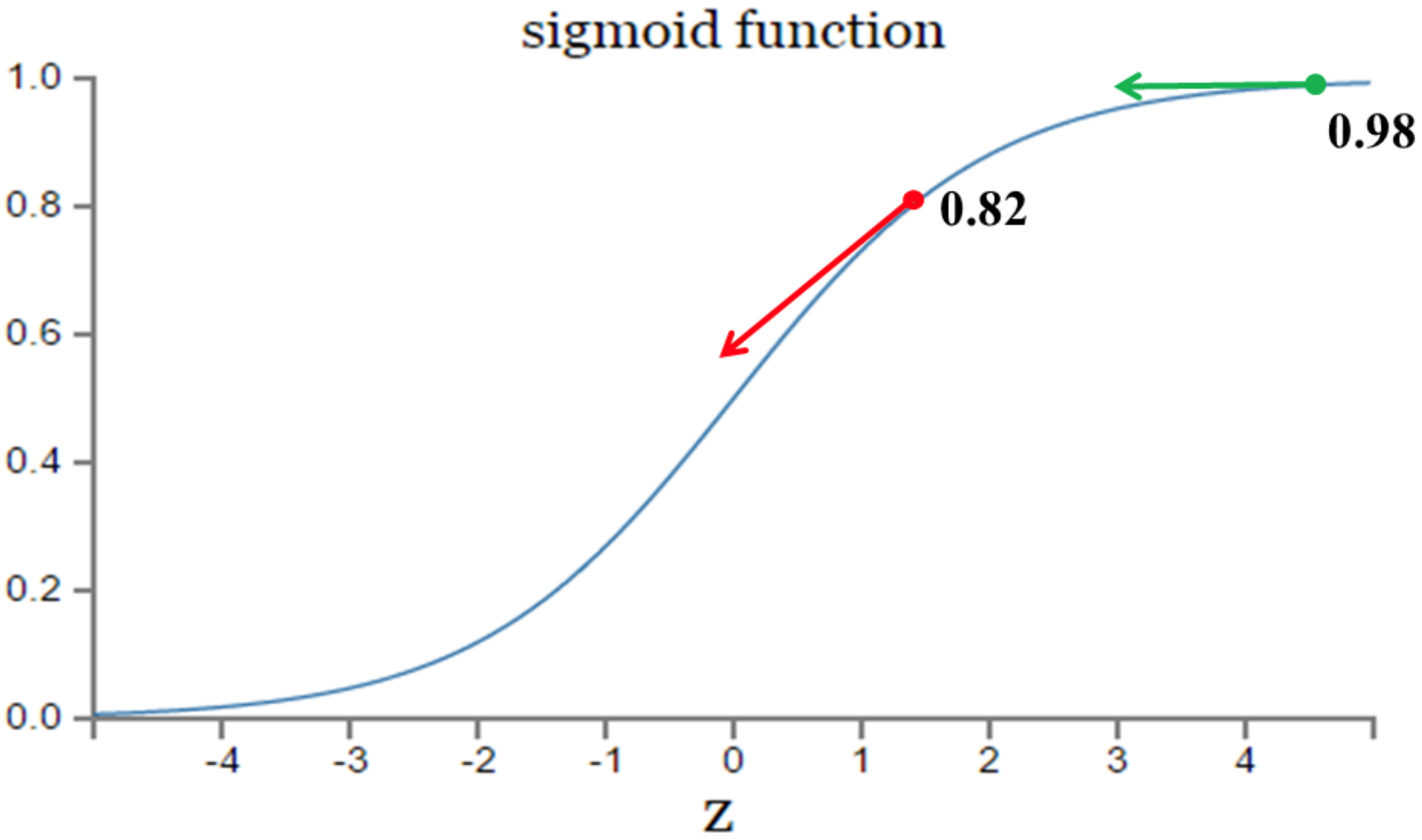
激活函数的梯度f’（z)越大，w的大小调整得越快，训练收敛得就越快。激活函数的梯度f’（z)越小，w的大小调整得越慢，训练收敛得就越慢。

二次代价函数

以一个二分类问题为例，进行两组实验。输入同一个样本数据x=1.0，该样本对应的分类为y=0，使用sigmoid激活函数。



二次代价函数



交叉熵（Cross-Entropy）

换一个思路，我们不改变激活函数，而是改变代价函数，
该用交叉熵代价函数：

$$E = -\left(t \ln y + (1-t) \ln (1-y)\right)$$

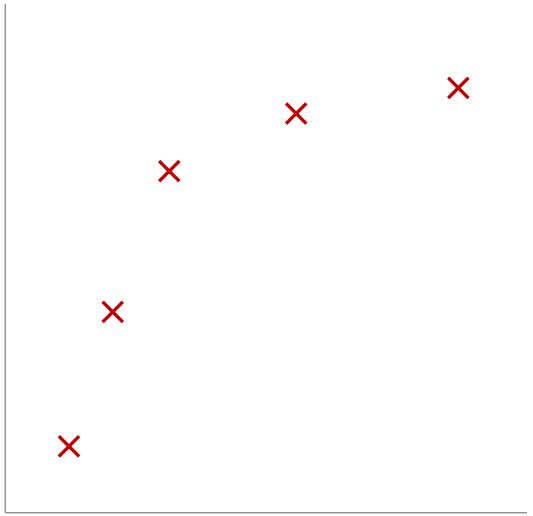
$$\begin{aligned}
\frac{\partial E}{\partial w} &= -\left(\frac{t}{f(z)} - \frac{1-t}{1-f(z)}\right)\frac{\partial f}{\partial w} \\
&= -\left(\frac{t}{f(z)} - \frac{1-t}{1-f(z)}\right) f'(z)\, x \\
&= \frac{f'(z)\, x}{f(z)(1-f(z))}(f(z)-t) \\
&= x(f(z)-t)
\end{aligned}$$

对于sigmoid函数：

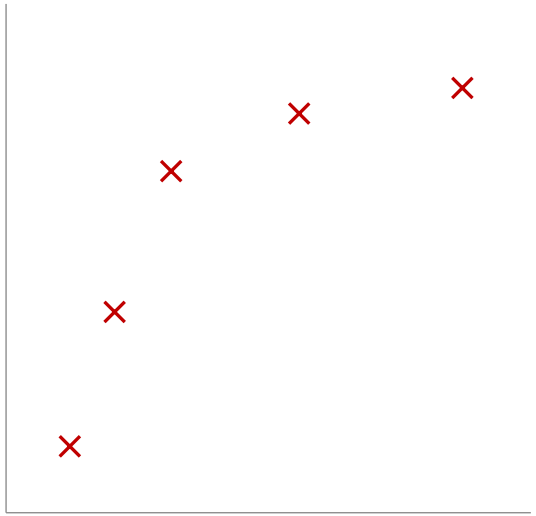
$$f'(z) = f(z)(1-f(z))$$

过拟合

回归拟合



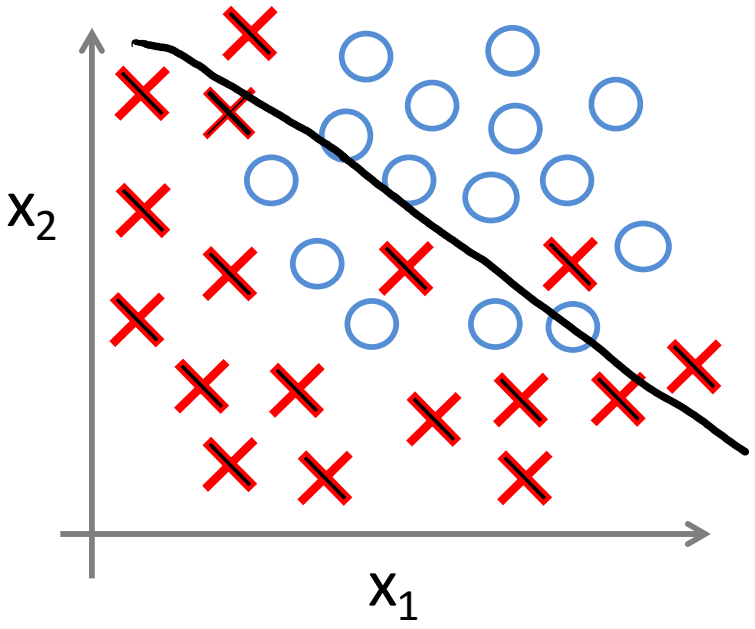
欠拟合(Underfitting)



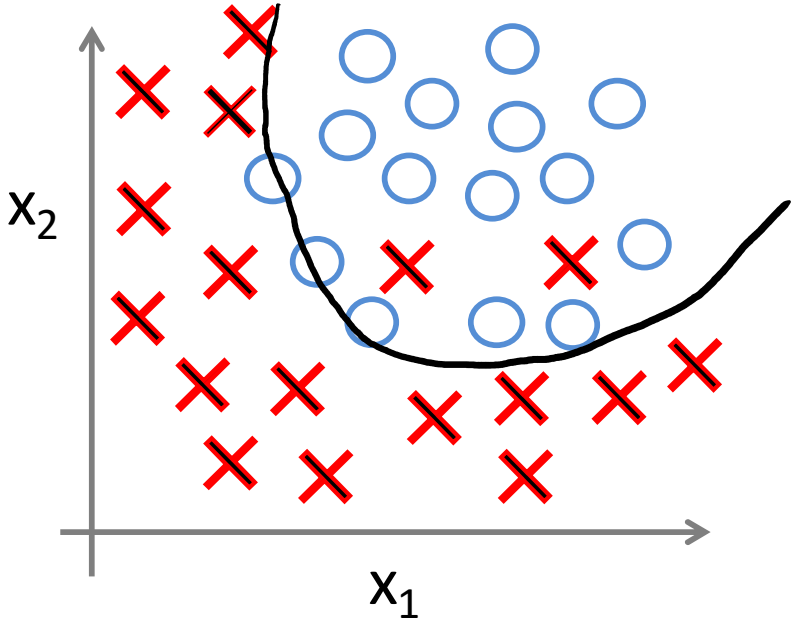
正确拟合(Just right)

过拟合(Overfitting)

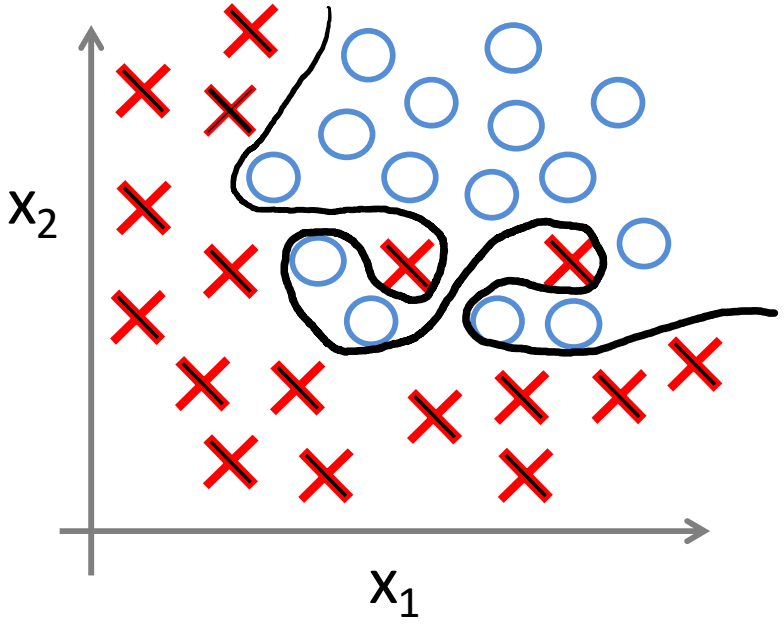
分类拟合



欠拟合(Underfitting)

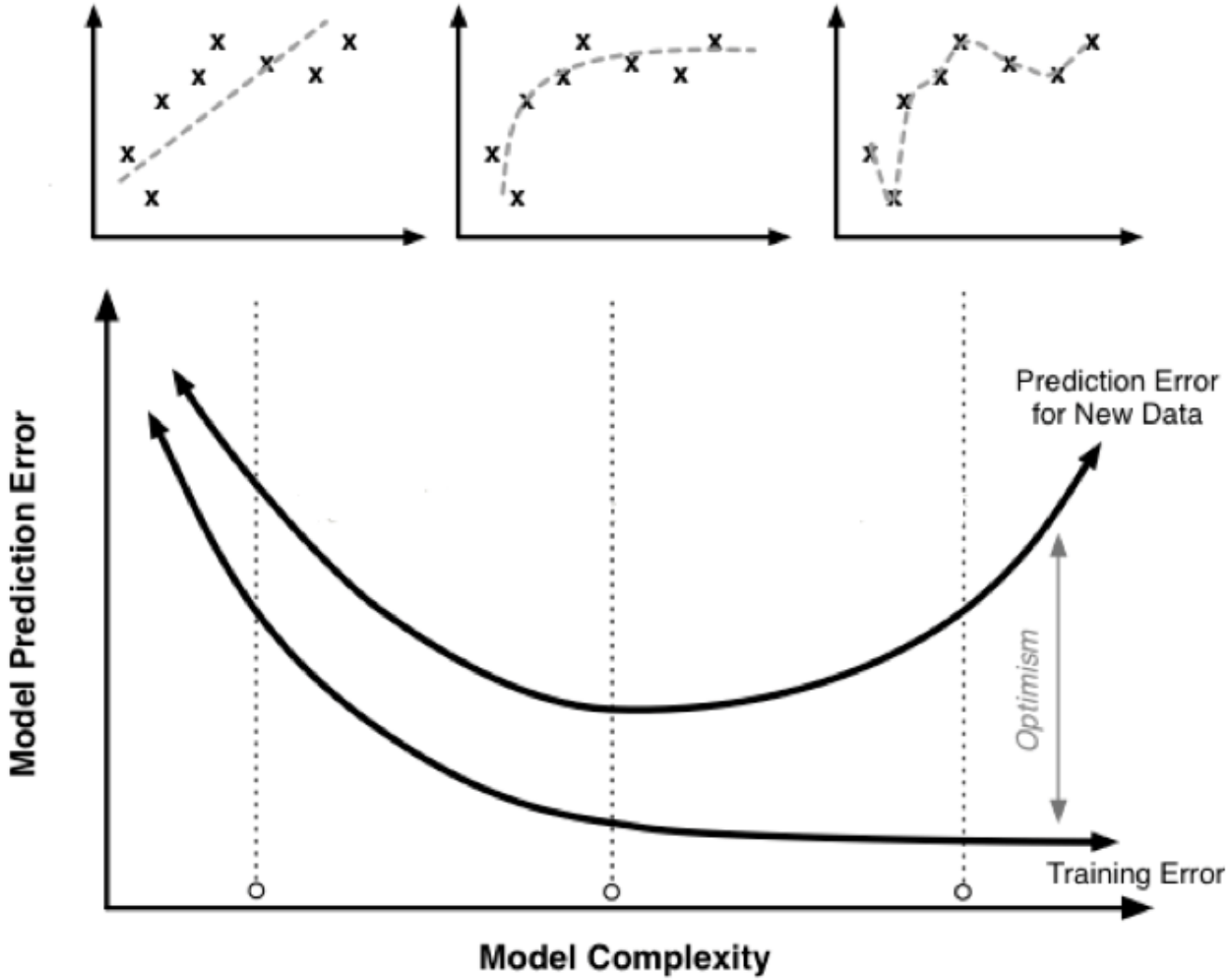


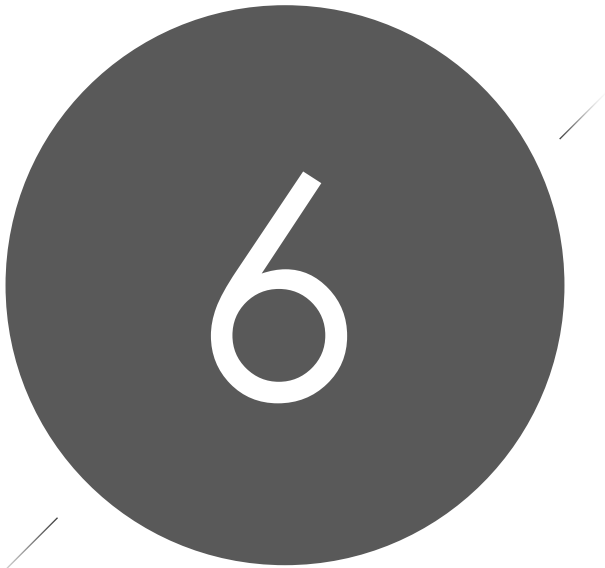
正确拟合(Just right)



过拟合(Overfitting)

过拟合导致测试误差变大





防止过拟合

增大数据集

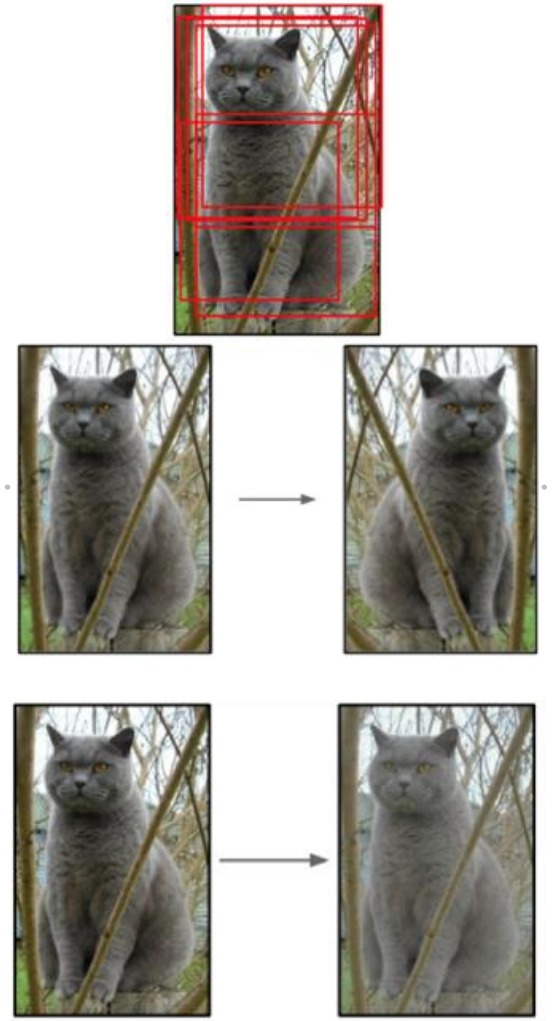
数据挖掘领域流行着这样一句话，“有时候拥有更多的数据胜过一个好的模型”。一般来说更多的数据参与训练，训练得到的模型就越好。如果数据太少，而我们构建的神经网络又太复杂的话就比较容易产生过拟合的现象。

增大图片数据集

1.随机裁剪

2.水平翻转

3.光照颜色抖动

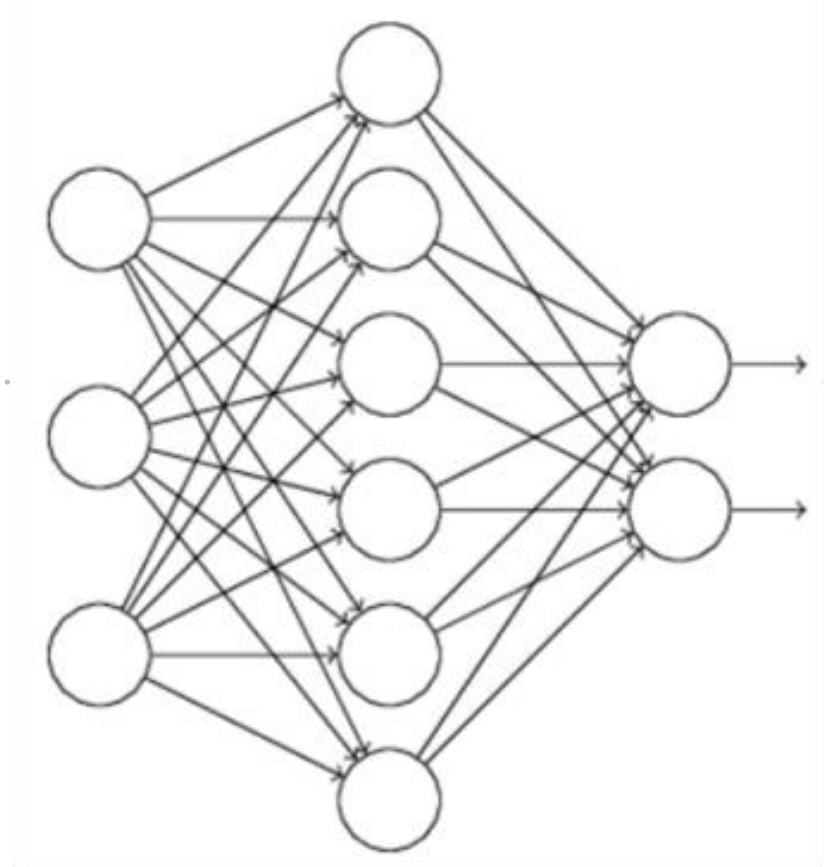


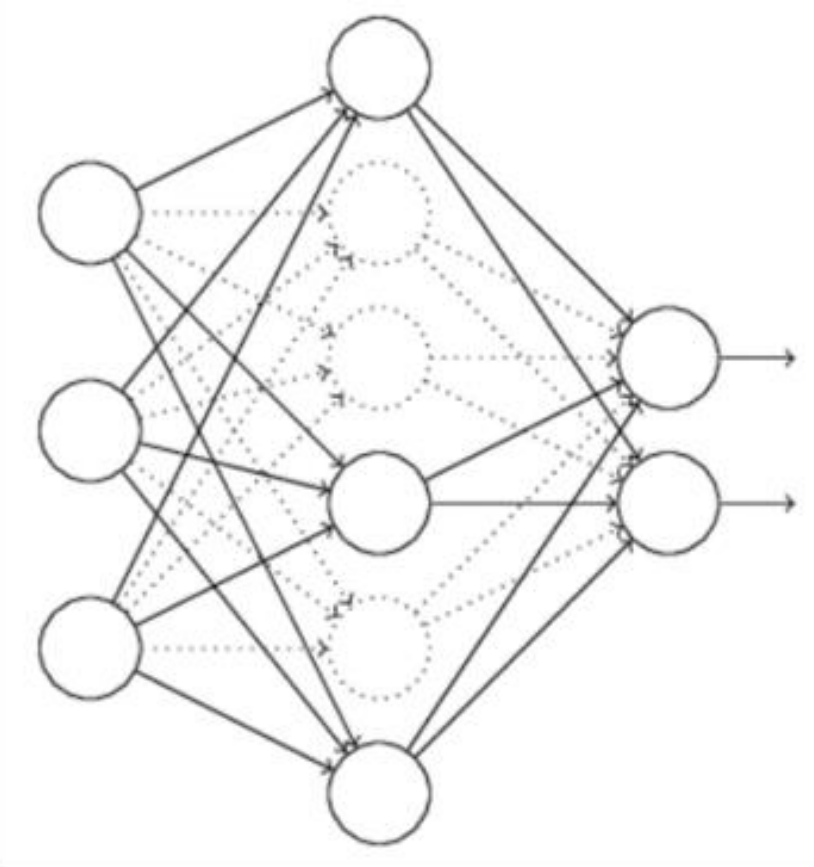
Early stopping

在训练模型的时候，我们往往会设置一个比较大的迭代次数。Early stopping便是一种提前结束训练的策略用来防止过拟合。

一般的做法是记录到目前为止最好的validation accuracy，当连续10个Epoch没有达到最佳accuracy时，则可以认为accuracy不再提高了。此时便可以停止迭代了（Early Stopping）。

Dropout





正则化项

C0代表原始的代价函数，n代表样本的个数，λ就是正则项系数，权衡正则项与C0项的比重。

L1正则化：

$$C = C_0 + \frac{\lambda}{n}\sum_w |w|$$

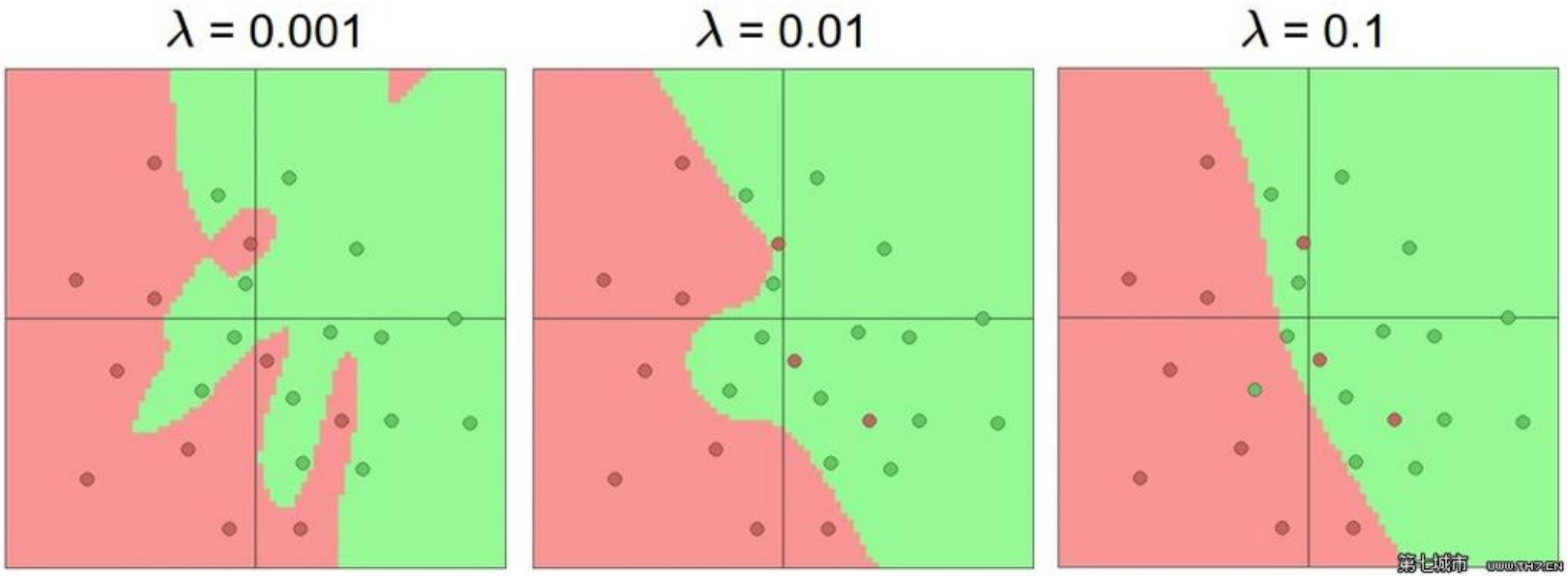
L1正则化可以达到模型参数稀疏化的效果

L2正则化：

$$C = C_0 + \frac{\lambda}{2n}\sum_w w^2$$

L2正则化可以使得模型的权值衰减，使模型参数值都接近于0。

正则化项



AI MOOC

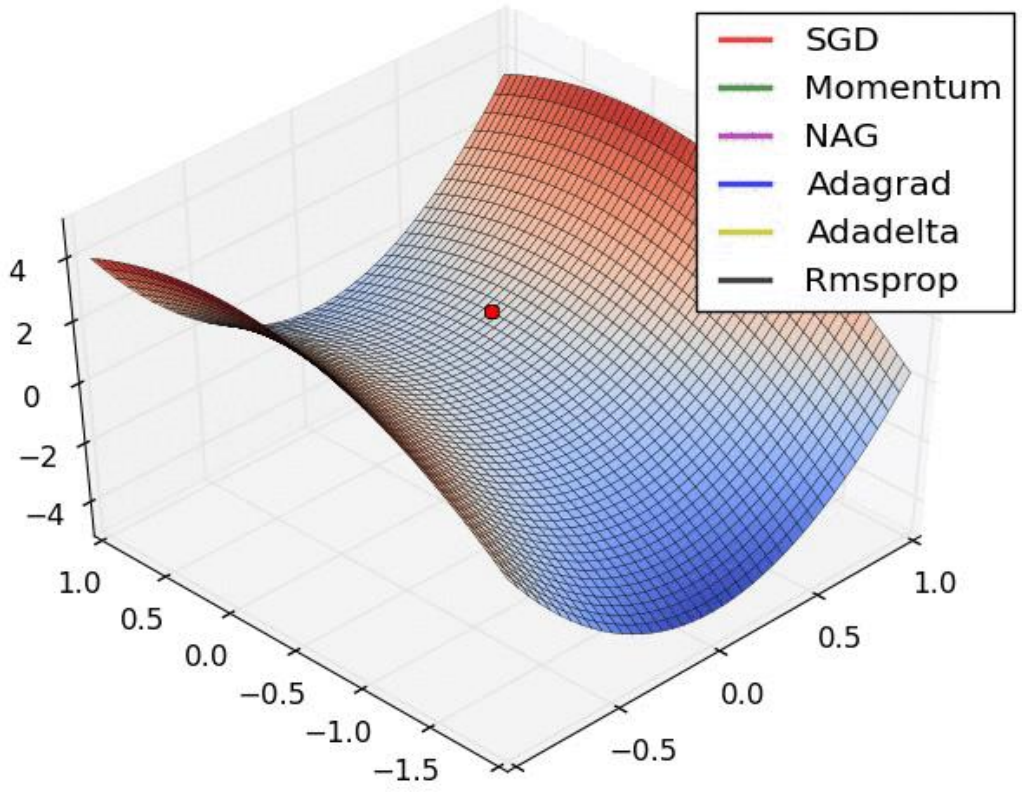
7

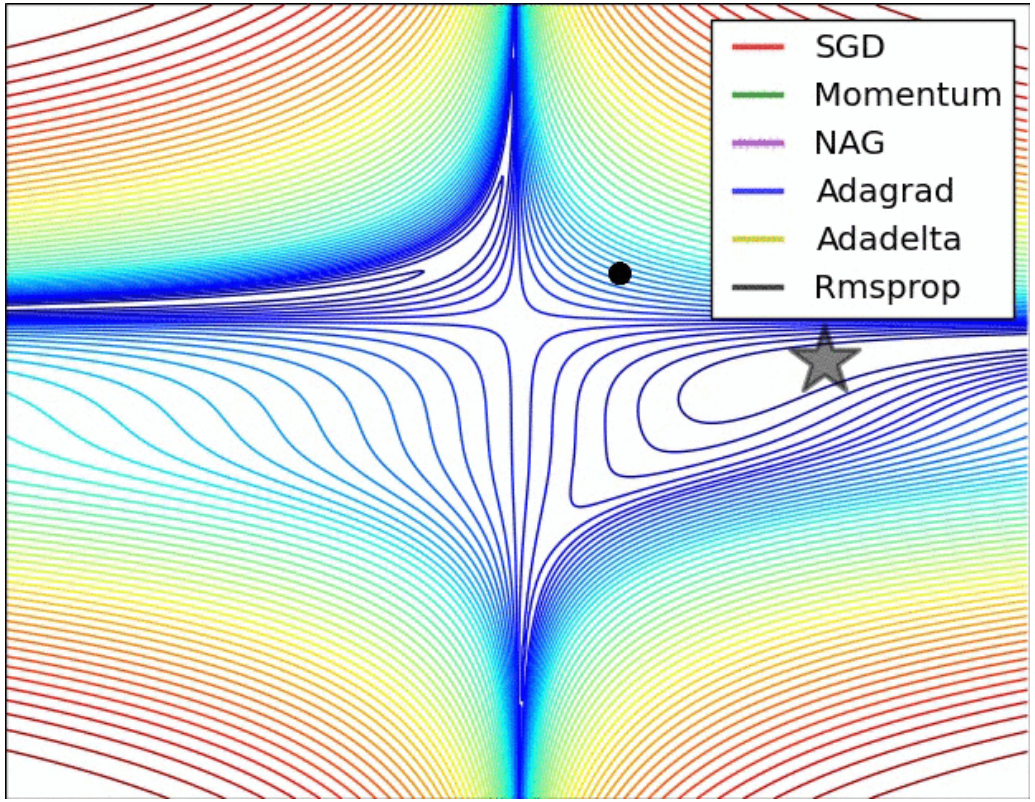
优化器

优化器

Adadelta
Adagrad
Adam
Adamax
AdamW
ASGD
LBFGS
RMSprop
Rprop
SGD
SparseAdam

优化器


SGD
Momentum
NAG
Adagrad
Adadelta
Rmsprop
4
2
0
−2
−4
1.0
0.5
0.0
−0.5
−1.0
−1.5
1.0
0.5
0.0
−0.5


SGD
Momentum
NAG
Adagrad
Adadelta
Rmsprop

AI MOOC

THANKS