智能之门

神经网络和深度学习入门

（基于Python的实现）

STEP 2 线性回归

第 6 章

多入单出的单层神经网络

线性二分类

6.1 线性二分类问题

6.2 二分类函数

6.3 线性二分类的实现

6.4 线性二分类的工作原理

6.5 逻辑与或非门

6.6 用双曲正切函数分类

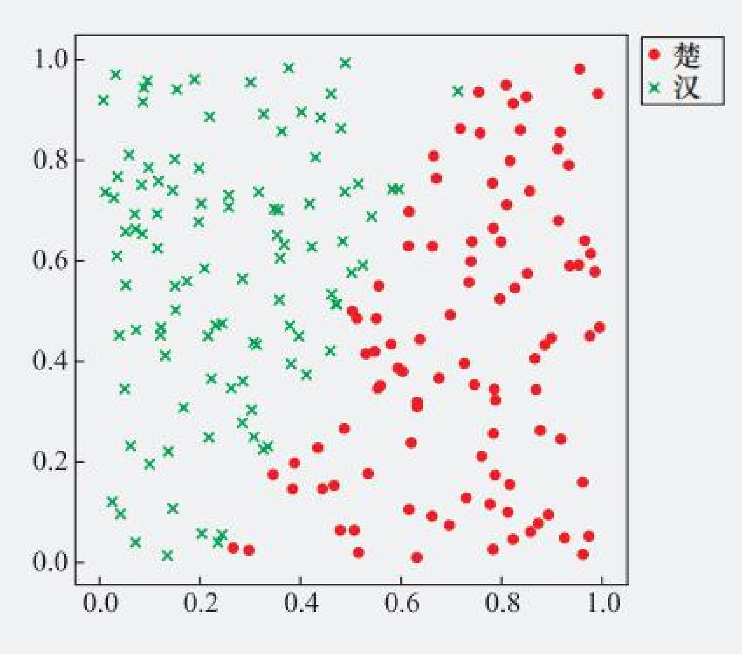
分类问题在很多资料中都称之为逻辑回归，其原因是使用了线性模型加一个 Logistic 二分类函数，共同构成了这样的分类器。神经网络的一个重要功能就是分类，现实世界中的分类任务复杂多样，但万变不离其宗，我们都可以用同一种模式的神经网络来处理。

本部分中，我们从最简单的线性二分类开始学习，包括其原理、实现、训练过程、推理过程等等，以可视化的方式来帮助大家更好地理解这些过程；并利用二分类知识，实现逻辑与门、与非门，或门，或非门。此外，我们还将探索双曲正切函数作为分类函数的可行性。

6.1 线性二分类问题

在中国象棋棋盘上，一道“楚河汉界”分隔开了两个阵营的棋子，其原型则来自公元前206年前后的楚汉战争。当时刘邦和项羽麾下的城池，在中原地区的地理位置示意图如右图所示；部分样本数据（已经过归一化）如下表所示。

样本序号	X_1：经度相对值	X_2：纬度相对值	Y：1= 汉 ,0= 楚
1	0.325	0.888	1
2	0.656	0.629	0
3	0.151	0.101	1
4	0.785	0.024	0
...	...	...	...
200	0.631	0.001	0

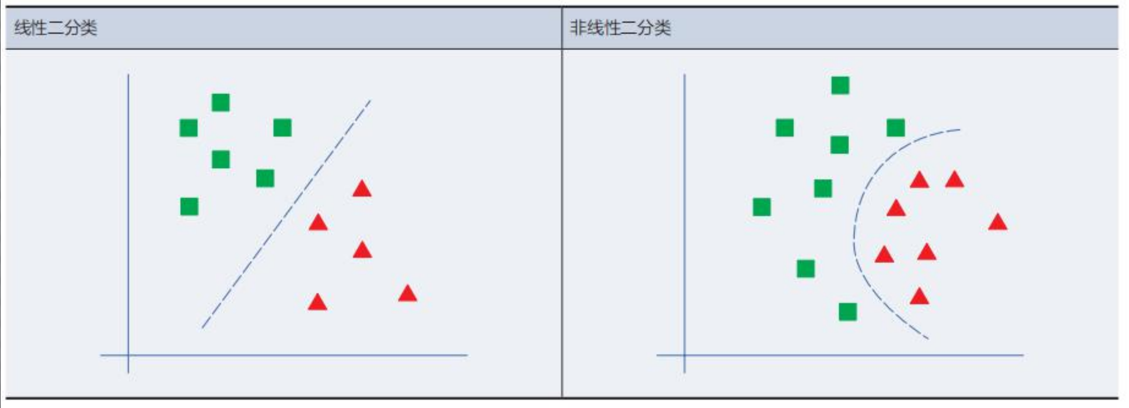


问题：经纬度相对坐标值分别为(0.58,0.92), (0.62,0.55), (0.39,0.29)时，属于楚还是汉？

6.1 线性二分类问题

➢ 逻辑回归模型

- 逻辑回归是用来计算“事件=Success”和“事件=Failure”的概率。当因变量的类型属于二元（1/0，真/假，是/否）变量时，我们就应该使用逻辑回归。
- 回忆线性回归，使用一条直线拟合样本数据，而逻辑回归是“拟合”0或1两个数值，而不是具体的连续数值，所以它叫广义线性模型。逻辑回归又称 *Logistic* 回归分析，常用于数据挖掘，疾病自动诊断，经济预测等领域。



逻辑回归的另外一个名字叫做分类器，分为线性分类器和非线性分类器，左图从直观上展示了二者的区别。本部分主要介绍线性分类器，并先介绍二分类器。

6.2 二分类函数

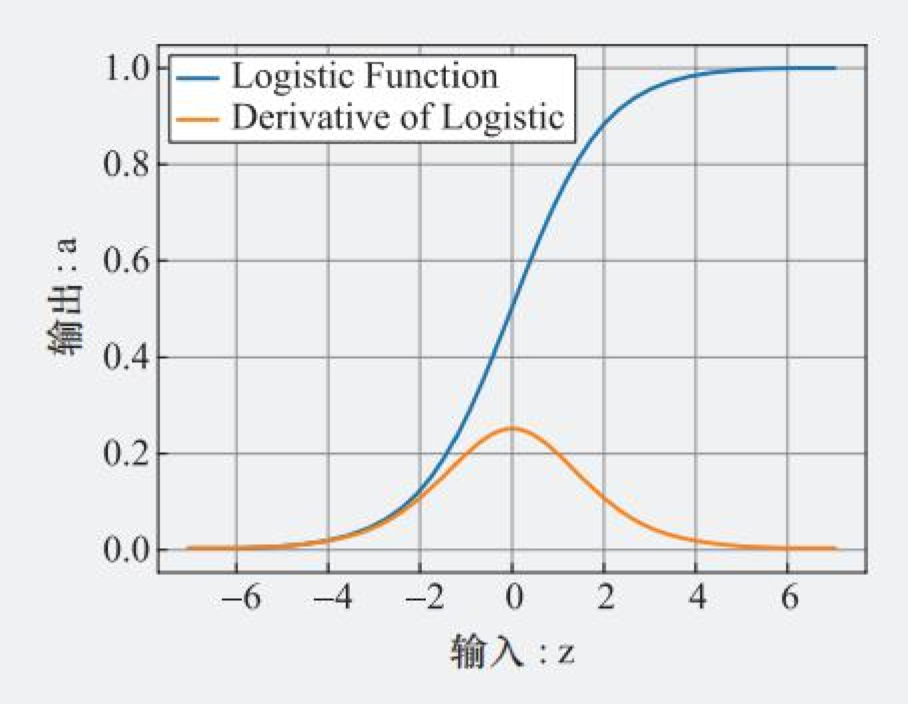
➢ 对率函数

- 对率函数，既可以作为激活函数使用，又可以当作二分类函数使用。在分类任务中，称其为 $Logistic$ 函数，而作为激活函数时，称其为 $Sigmoid$ 函数。它的定义域为全体实数，值域为 $(0,1)$。

$$Logistic(z) = \frac{1}{1+e^{-z}}$$

- 记 $a = Logistic(z)$，则有以下导数公式成立，值域为 $(0,0.25]$：

$$Logistic'(z) = a(1-a)$$



6.2 二分类函数

- 训练时，一个样本 x 在经过神经网络的最后一层的矩阵运算结果作为输入 z，经过 $Logistic$ 计算后，输出一个 $(0,1)$ 之间的预测值。我们假设这个样本的标签值为 0 属于负类，如果其预测值越接近 0，就越接近标签值，那么误差越小，反向传播的力度就越小。
- 推理时，我们预先设定一个阈值比如 0.5，则当推理结果大于 0.5时，认为是正类；小于 0.5 时认为是负类；等于 0.5 时，根据情况自己定义。阈值也不一定就是 0.5 ，也可以是 0.65 等等，阈值越大，准确率越高，召回率越低；阈值越小则相反，准确度越低，召回率越高。

6.2 二分类函数

➢ 前向计算

- 矩阵运算：

$$z = xw + b$$

- 分类计算：

$$a = Logistic(z) = \frac{1}{1 + e^{-z}}$$

- 交叉熵损失函数：

$$loss(w, b) = -[y \ln a + (1 - y) \ln(1 - a)]$$

6.2 二分类函数

➢ 反向传播

- 求误差对 a 的偏导:

$$\frac{\partial loss}{\partial a}=-\left(\frac{y}{a}-\frac{1-y}{1-a}\right)=-\frac{y(1-a)-(1-y)a}{a(1-a)}=\frac{a-y}{a(1-a)}$$

- 用链式法则求误差对 z 的偏导:

$$\frac{\partial loss}{\partial z}=\frac{\partial loss}{\partial a}\cdot\frac{\partial a}{\partial z}=a-y$$

- 多样本情形:

$$\frac{\partial J}{\partial Z}=\begin{pmatrix}\frac{\partial J}{\partial z_1} & \cdots & \frac{\partial J}{\partial z_n}\end{pmatrix}^T=(a_1-y_1 \quad \cdots \quad a_n-y_n)^T=A-Y$$

➢ 匹配关系

- 损失函数满足二分类的要求，无论是正例还是反例，都是单调的;
- 损失函数可导，以便于使用反向传播算法;
- 计算过程非常简单，一个减法就可以搞定。

6.2 二分类函数

➢ 对数几率的来历

- 假设有一个硬币，抛出落地后，得到正面的概率是 a，得到反面的概率是 $1-a$。如果用正面的概率除以反面的概率，得到的这个数值叫做ODDS，即几率。它反映了一个样本作为正例的相对可能性：

$$odds = \frac{a}{1-a}$$

- 对几率取对数，即成为对数几率

$$\ln(odds) = \ln\frac{a}{1-a}$$

6.2 二分类函数

概率 a	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
反概率 $(1-a)$	1	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0
几率 *odds*	0	0.11	0.25	0.43	0.67	1	1.5	2.33	4	9	∞
对数几率 ln(*odds*)	N/A	−2.19	−1.38	−0.84	−0.4	0	0.4	0.84	1.38	2.19	N/A

- 从以上概率到对数几率的表中可见，对数几率的值近似呈线性关系，因而应用线性模型：

$$\ln\frac{a}{1-a} = xw + b$$

$$\frac{a}{1-a} = e^{xw+b}$$

$$a = \frac{1}{1+e^{-(xw+b)}} = \frac{1}{1+e^{-z}}$$

6.2 二分类函数

以上推导过程，实际上就是用线性回归模型的预测结果来逼近样本分类的对数几率。这就是它虽然称作逻辑回归，但其实是分类学习方法的原因。

➢ 这种方法的优点如下：

- 直接对分类可能性建模，无需事先假设数据分布，避免了假设分布不准确所带来的问题；
- 不仅预测出类别，而是得到了近似的概率，这对许多需要利用概率辅助决策的任务很有用；
- 对率函数是任意阶可导的凸函数，有很好的数学性，许多数值优化算法都可以直接用于求取最优解。

6.3 线性二分类的实现

看楚汉城池示意图，在两个颜色区域之间似乎存在一条分割的直线，即线性可分的。

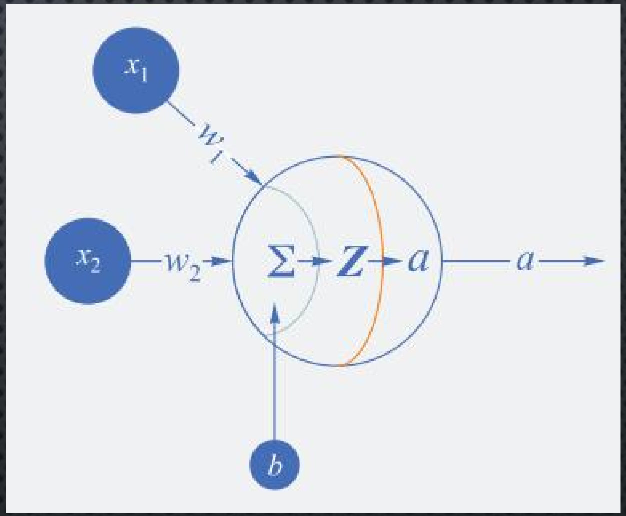
➢ 神经网络结构

- 从视觉上判断是线性可分的，使用单层神经网络；
- 输入层设置两个输入单元，表示经纬度：$X = (x_1, x_2)$。
- 输出层设置一个单元，表示地盘所属阵营：

$$z = XW + B, a = Logistic(z)$$

- 权重参数为：$W = (w_1, w_2)^T, B = (b)$。
- 损失函数为交叉熵损失函数，反向传播公式为：

$$\frac{\partial J}{\partial W} = X^T(A - Y), \qquad \frac{\partial J}{\partial B} = A - Y$$

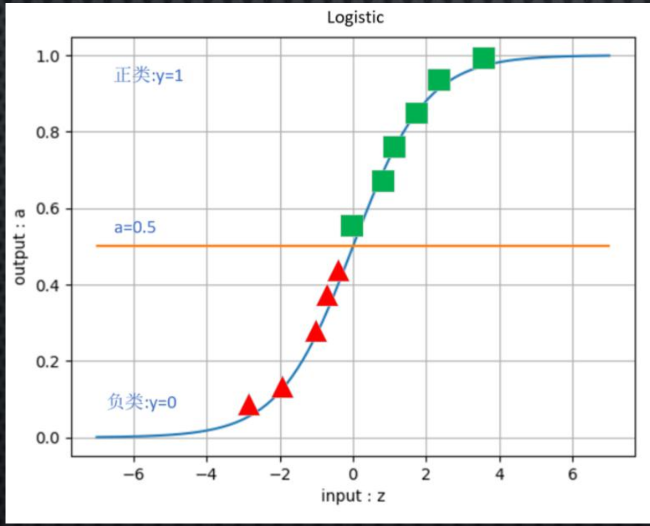


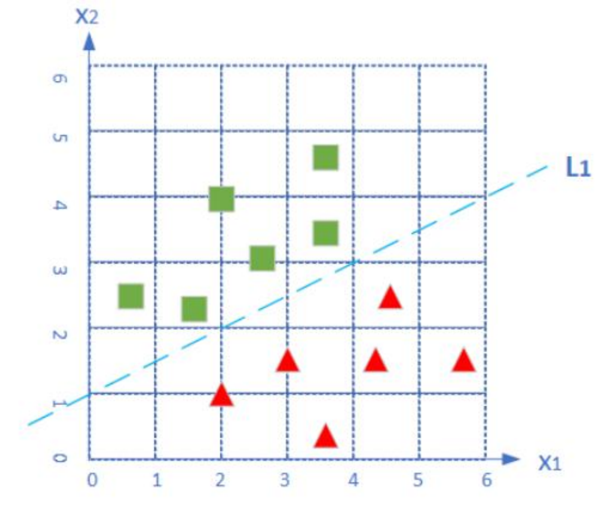
分类结果：经纬度相对坐标值为$(0.58,0.92), (0.62,0.55), (0.39,0.29)$时，分别属于汉、楚、汉。

6.4 线性二分类的工作原理

回忆一下前面学习过的线性回归，通过用均方差函数的误差反向传播的方法，不断矫正拟合直线的斜率和截距，因为均方差函数能够准确地反映出当前的拟合程度。

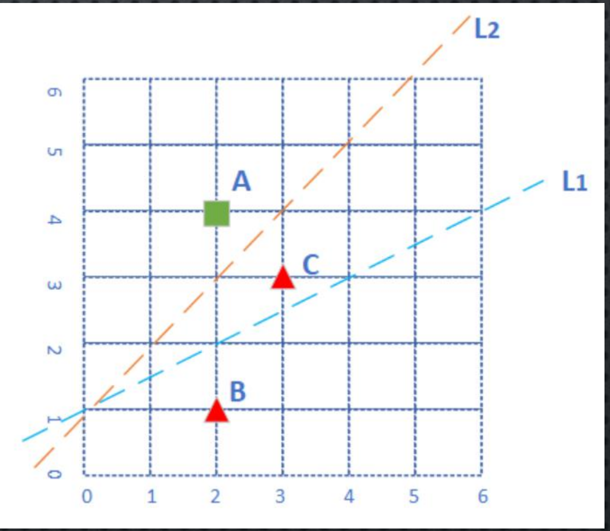
线性分类与线性回归相似的地方是，两者都需要划出那条“直线”来，但是不同的地方也很明显：线性回归要求用直线来拟合所有样本，使得各个样本到这条直线的距离尽可能最短；而线性分类则要求用直线来分割所有样本，使得正例样本和负例样本尽可能分布在直线两侧。





6.4 线性二分类的工作原理

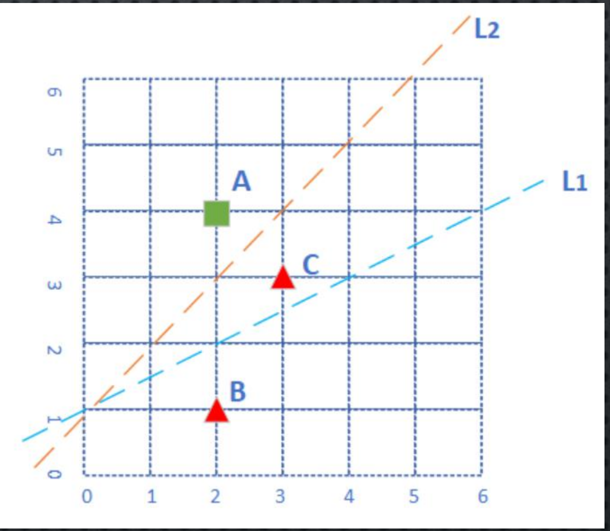
- **分类线为L_1时**
  - 此时有：$w_1 = -1, w_2 = 2, b = -2$。
  - 前向计算：$z_A = 4, z_B = -2, z_C = 1$。样本 A, B 的分类正确，样本 C 的分类错误。
  - 样本损失：$loss_A = 0.018, loss_B = 0.112, loss_C = 1.313$。
  - 样本 C 的误差较大，反向传播力度也大。



6.4 线性二分类的工作原理

➢ **分类线为L_2时**

- 此时有：$w_1 = -1, w_2 = 1, b = -1$。
- 前向计算：$z_A = 1, z_B = -2, z_C = -1$。样本均分类正确。
- 样本损失：$loss_A = 0.018, loss_B = 0.112, loss_C = 1.313$。
- 样本 C 的误差较大，反向传播力度也大。

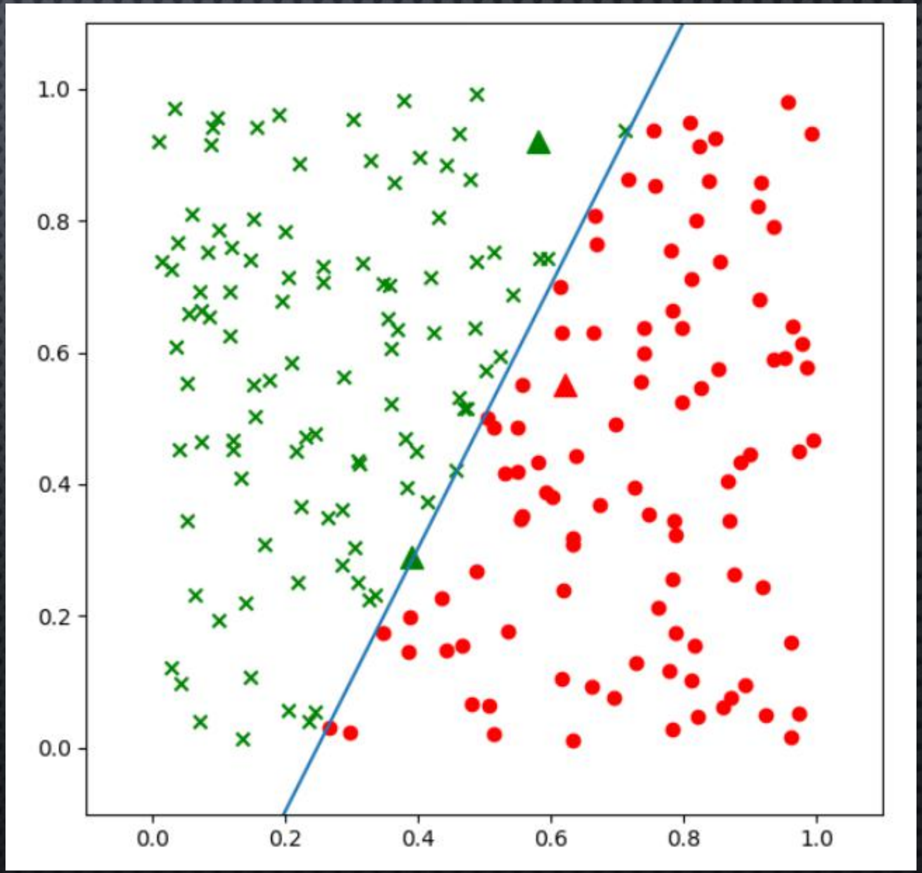


$Logistic$函数存在于分类任务中的意义：告知神经网络误差力度有多大，使之进行有效的反向传播。

6.4 线性二分类的工作原理

➢ 最终运行结果可视化

- 收敛后的分类结果如右图所示，分类直线已经基本可以分开不同类别的样本。

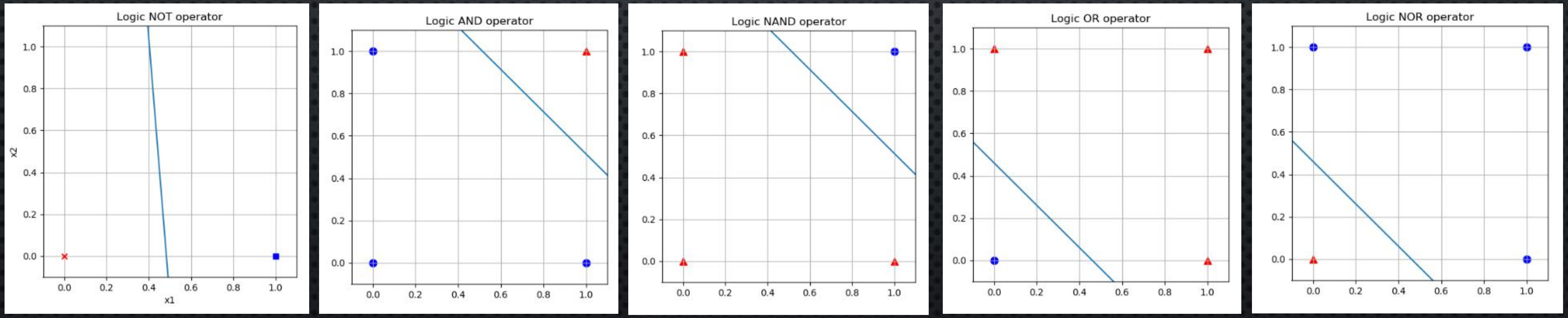


6.5 逻辑与或非门

➢ 逻辑运算法则与分类结果展示

- 右表为不同样本经过逻辑运算的结果，下图从左到右依次表示非门、与门、与非门、或门、或非门的分类结果，说明神经网络可以很好地胜任此类型的任务，找到完美的分割线。

样本	x1	x2	逻辑与y	逻辑与非y	逻辑或y	逻辑或非y
1	0	0	0	1	0	1
2	0	1	0	1	1	0
3	1	0	0	1	1	0
4	1	1	1	0	1	0

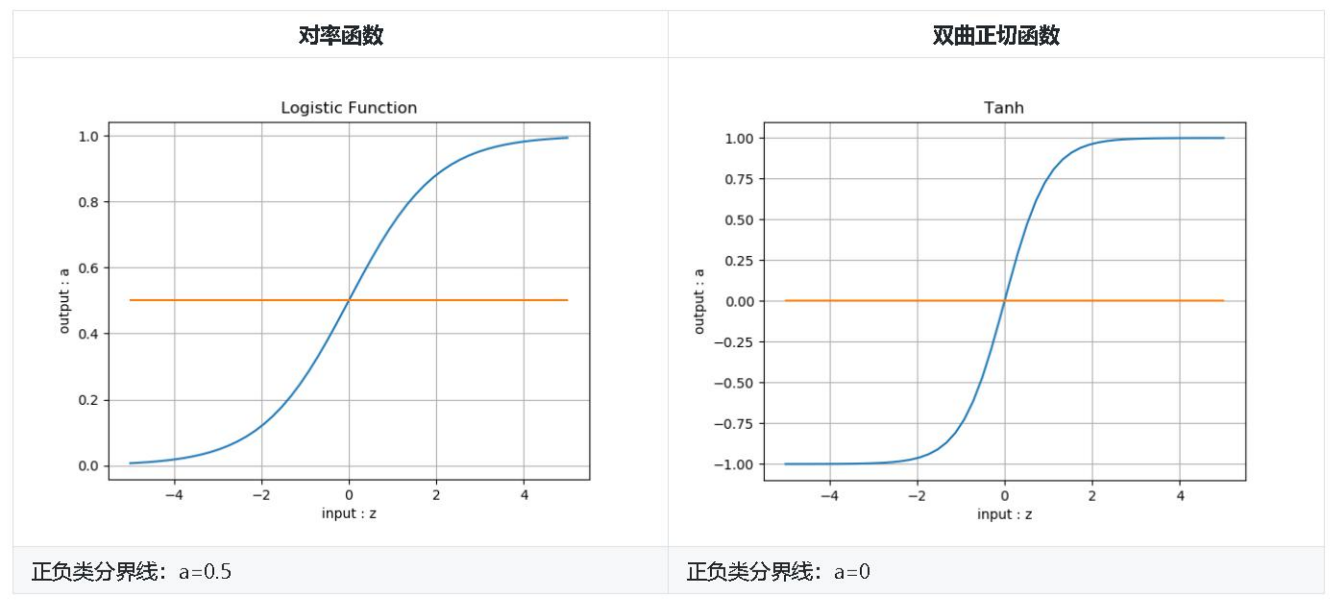


6.6 用双曲正切函数分类

➢ 双曲正切函数

$$Tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} = \frac{2}{1 + e^{-2z}} - 1 = 2 \cdot Logistic(2z) - 1$$

从函数形式上看，只要正确修改反向传播梯度公式和损失函数公式，双曲正切函数亦可作为分类函数。



THE END

谢谢！