



李哥考研

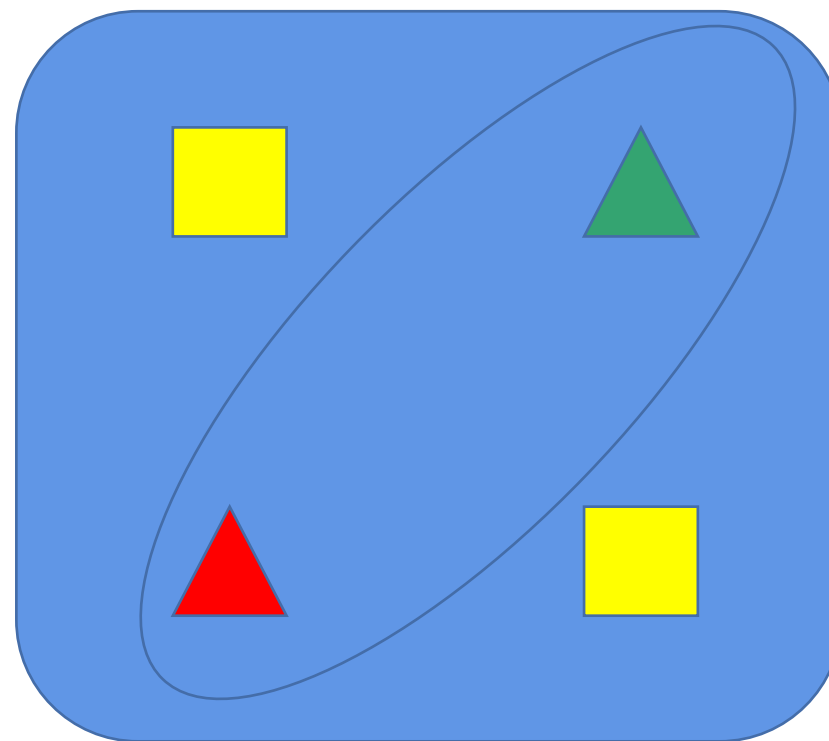
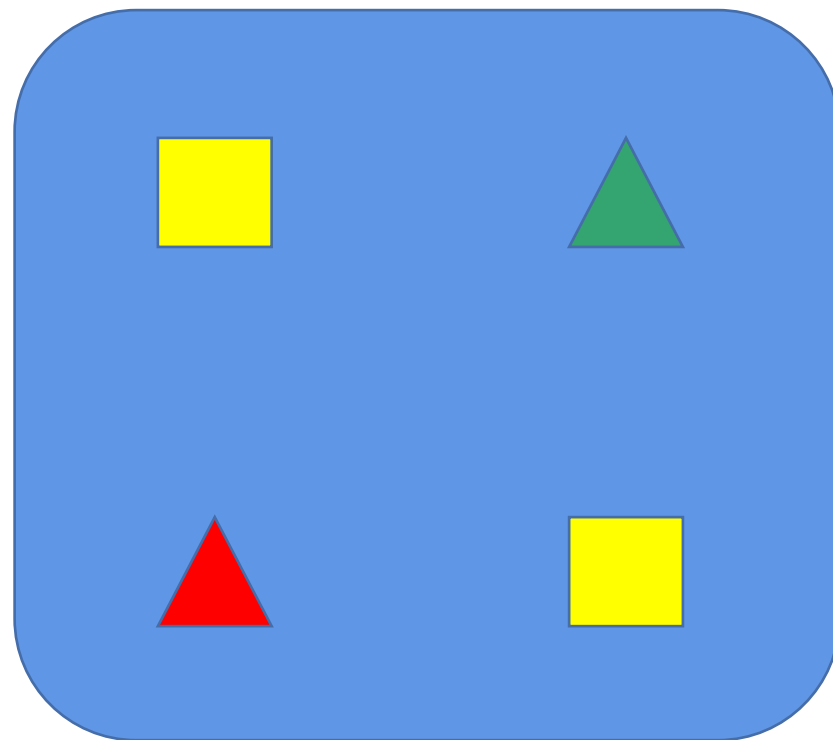
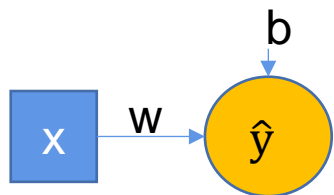
# 第一章

# 多层神经网络

# 线性函数与多层神经元



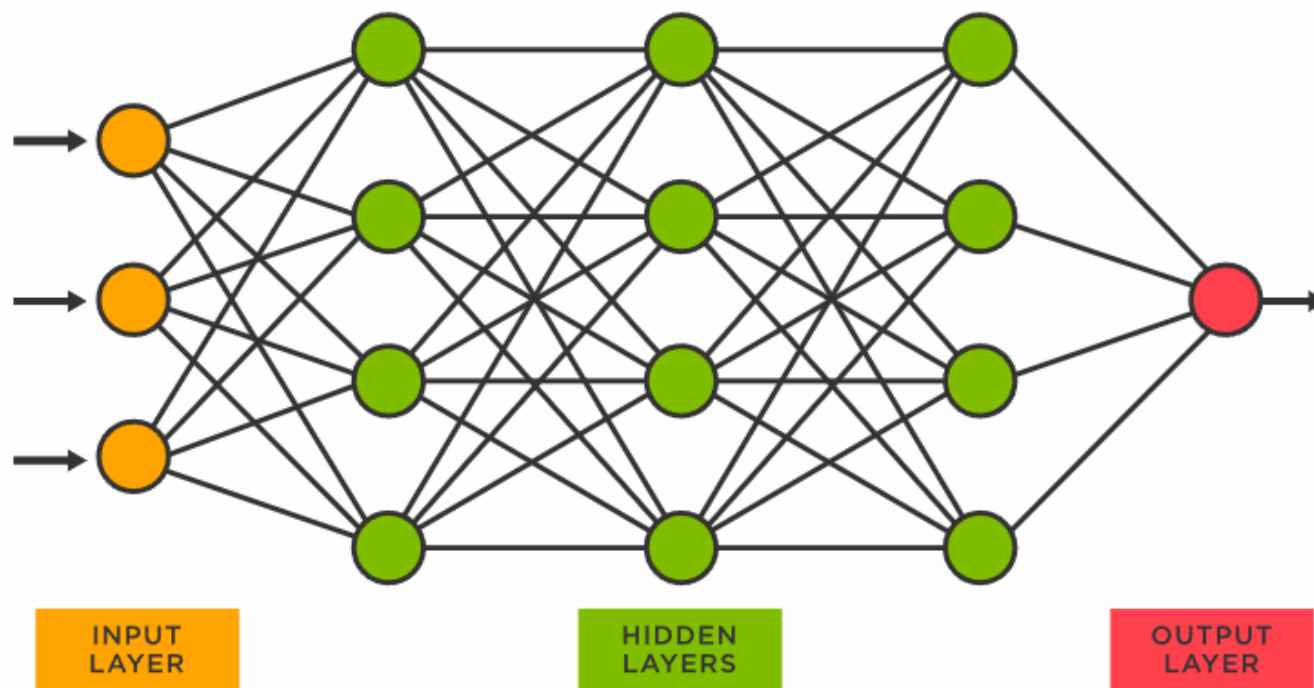
李哥考研

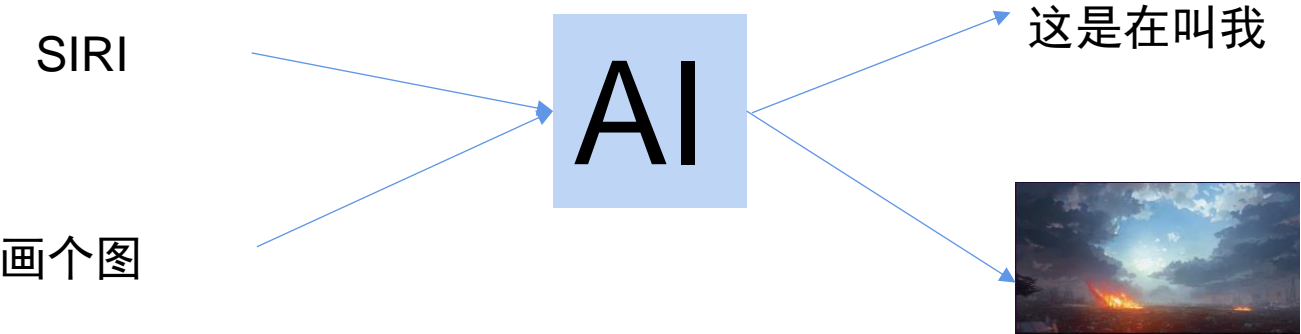
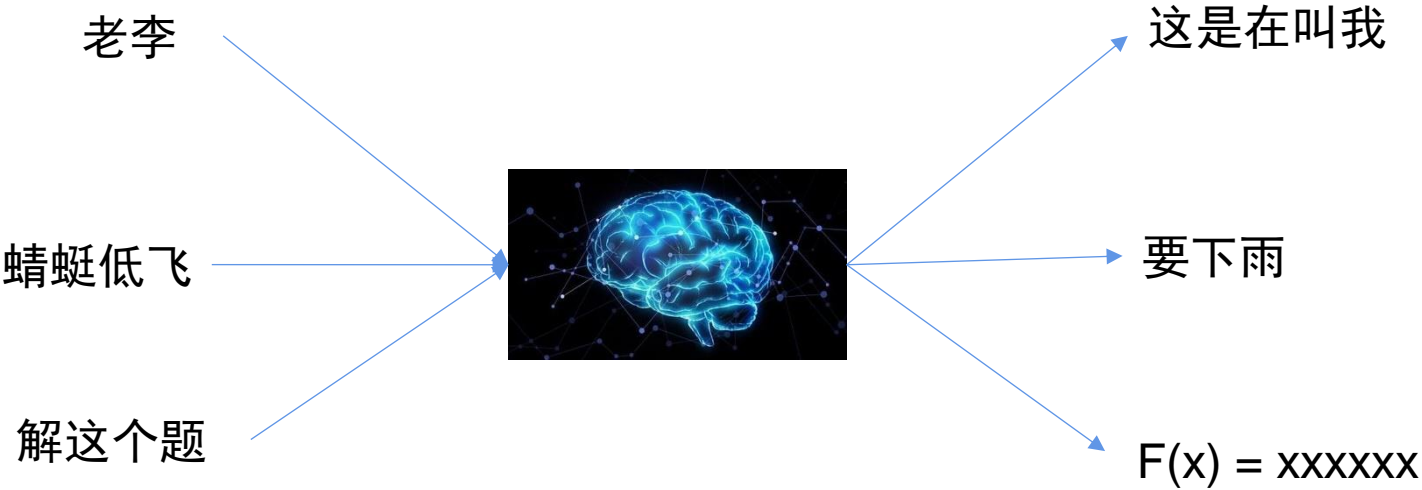


# 更复杂的神经网络



李哥考研

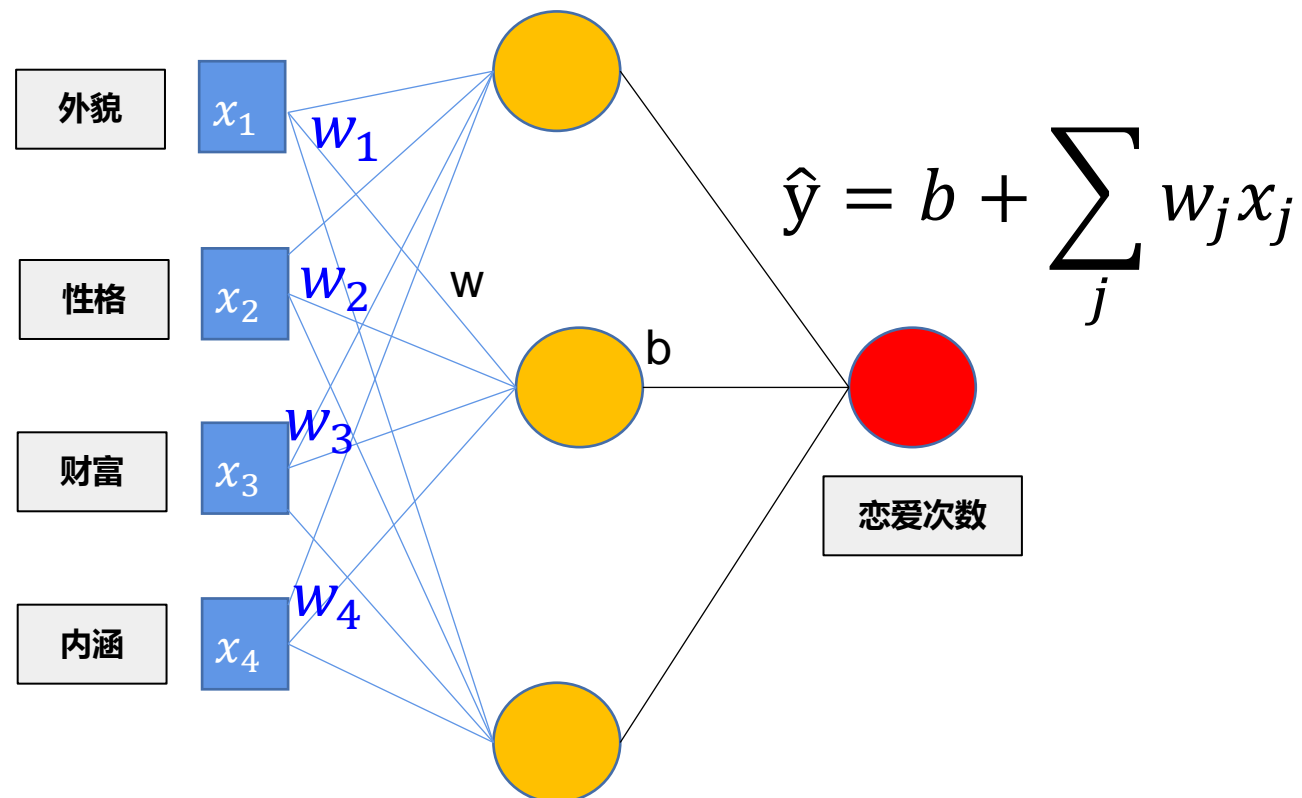
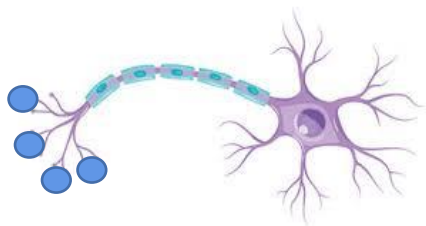
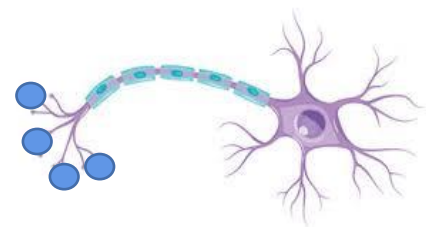
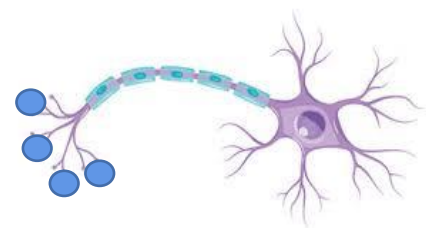




# 人类大脑神经与神经元



李哥考研

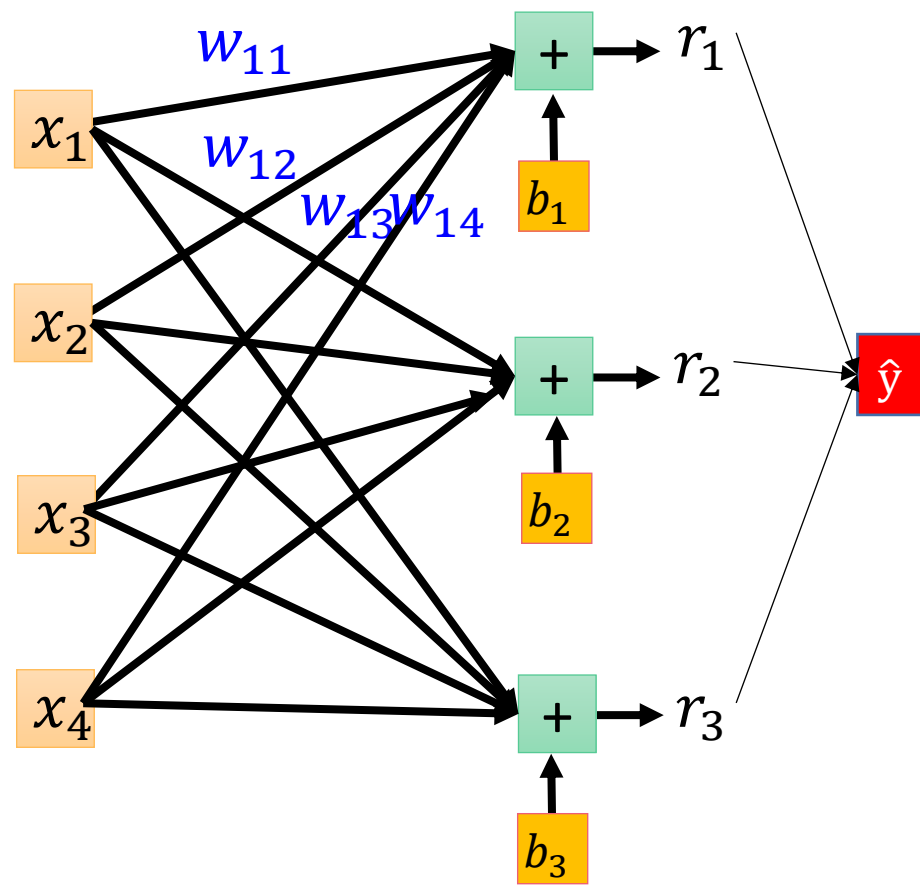
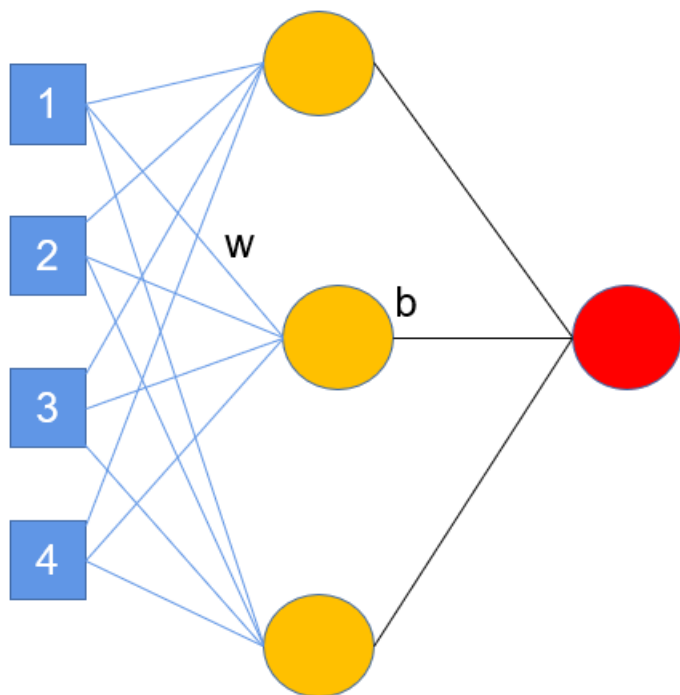


# 神经元与矩阵



李哥考研

$$r_1 = b_1 + w_{11}x_1 + w_{12}x_2 + w_{13}x_3 + w_{14}x_4$$



# 神经元与矩阵



李哥考研

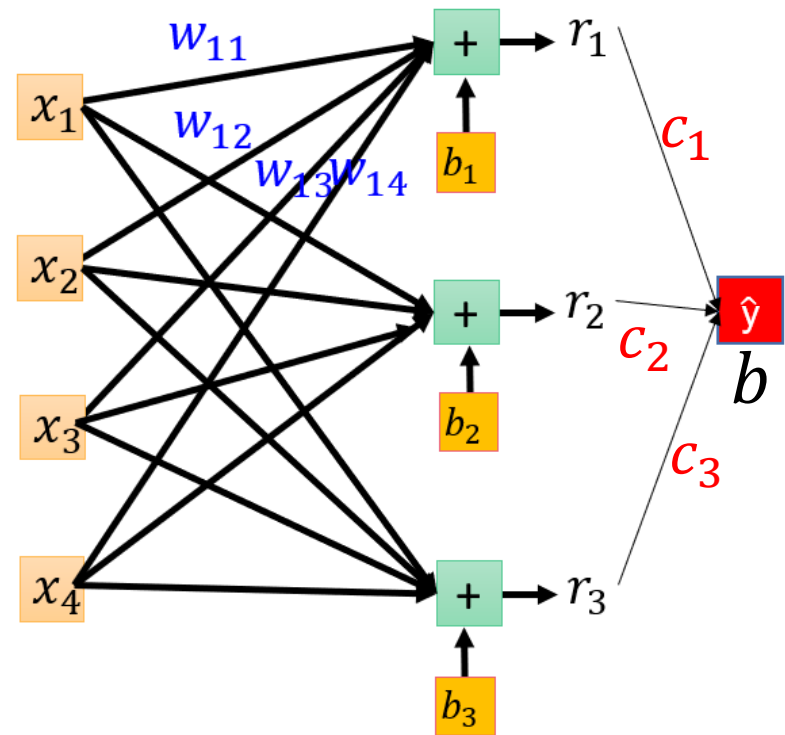
$$r_1 = b_1 + w_{11}x_1 + w_{12}x_2 + w_{13}x_3 + w_{14}x_4$$

$$r_2 = b_2 + w_{21}x_1 + w_{22}x_2 + w_{23}x_3 + w_{24}x_4$$

$$r_3 = b_3 + w_{31}x_1 + w_{32}x_2 + w_{33}x_3 + w_{34}x_4$$

$$\begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} + \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

$$\mathbf{r} = \mathbf{b} + \mathbf{W} \mathbf{x}$$

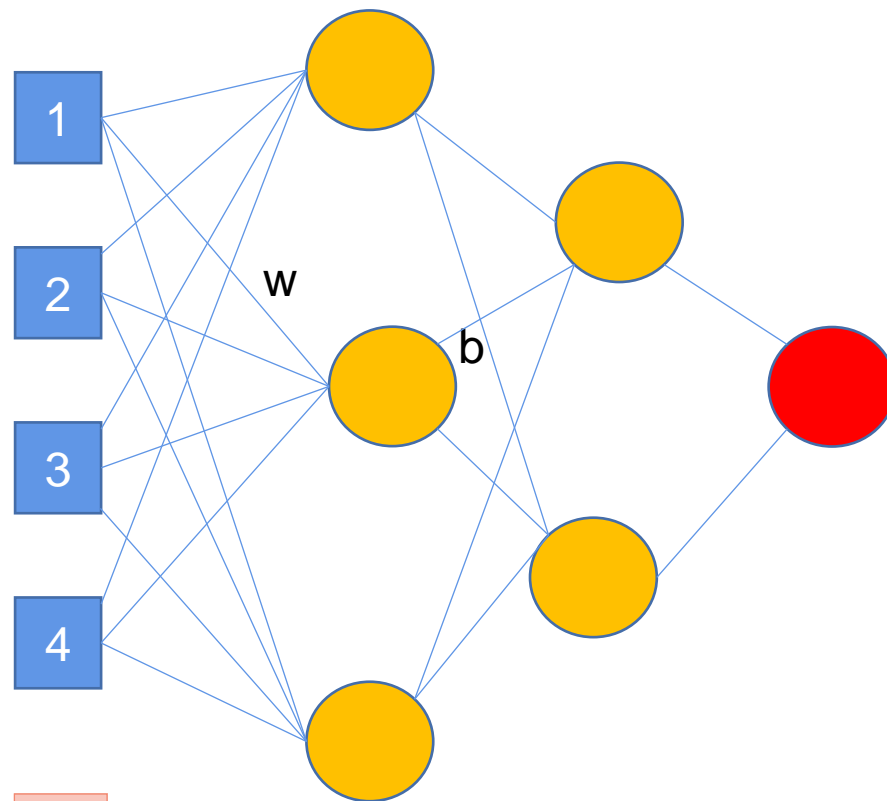
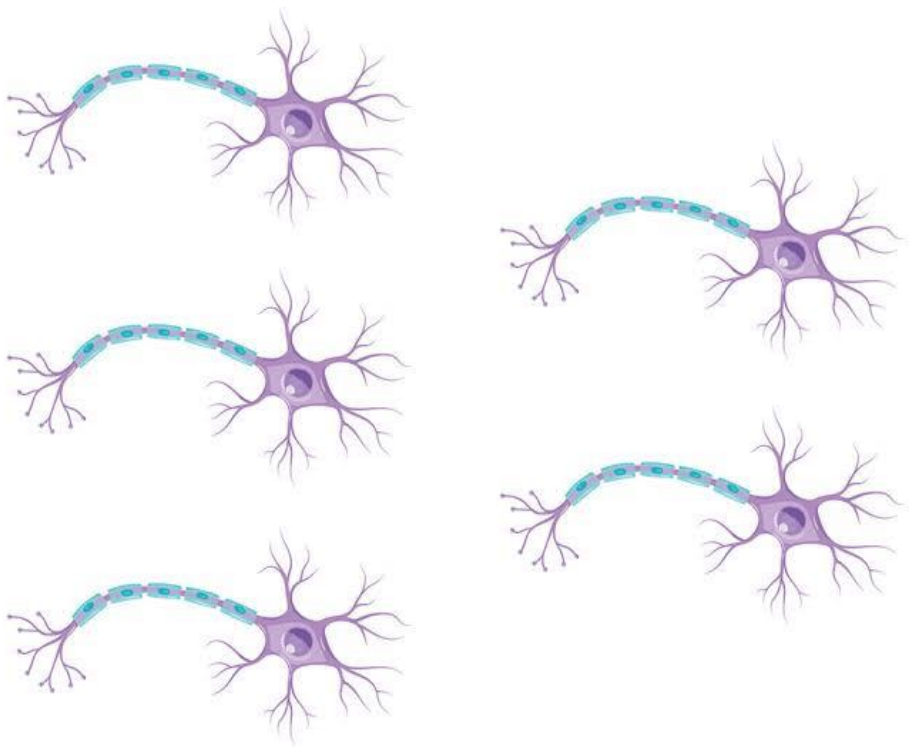


$$\hat{y} = \mathbf{b} + \mathbf{c}^T \mathbf{r}$$

# 神经元的串联



李哥考研



$$\hat{y} = b + c^T (b + W r)$$



# 这样有什么问题？



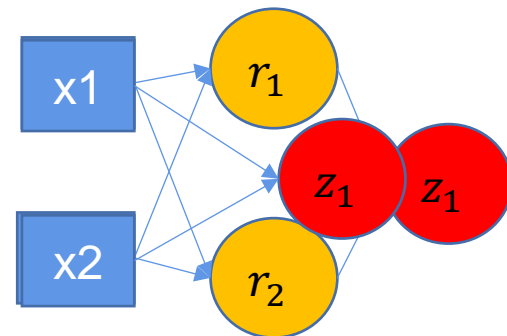
李哥考研

串联的神经元似乎只有传递的作用，  
那么 一根和多根似乎没有区别。

$$r_1 = b_1 + w_{11}x_1 + w_{12}x_2$$

$$r_2 = b_2 + w_{21}x_1 + w_{22}x_2$$

$$z_1 = b_1 + w_{11}r_1 + w_{12}r_2$$



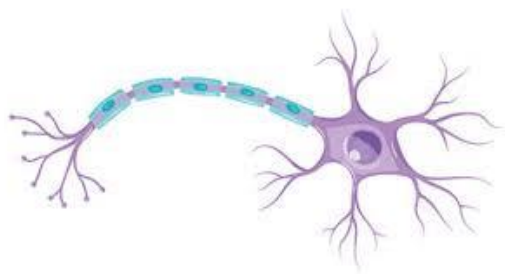
$$z_1 = b_1 + w_{11}(b_1 + w_{11}x_1 + w_{12}x_2) + w_{12}(b_2 + w_{21}x_1 + w_{22}x_2)$$

$$= b_1 + \alpha_1 x_1 + \alpha_2 x_2$$

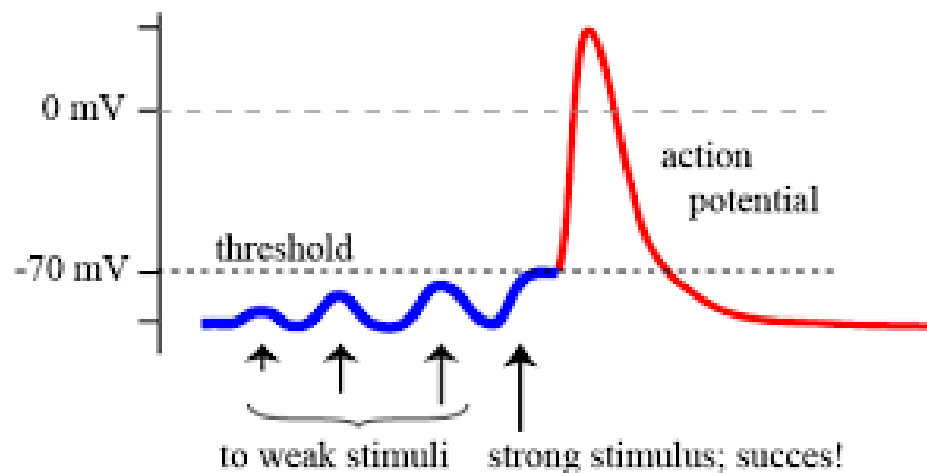
无论多少层都只有一层



生物神经元具有兴奋和抑制两种状态，当接受的刺激高于一定阈值时，则会进入兴奋状态并将神经冲动由轴突传出，反之则没有神经冲动。



我有自己的决定权

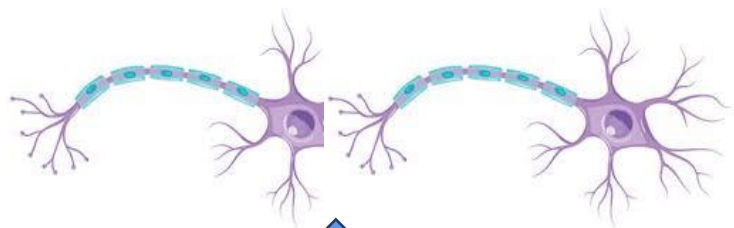


激活函数

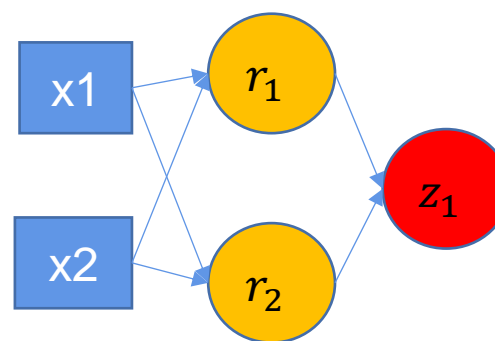
# 激活函数的位置



李哥考研



刺激阈值



$$r_1 = \text{sigmoid}(b_1 + w_{11}x_1 + w_{12}x_2)$$

激活函数

# 激活函数和非线性因素。



李哥考研

- 如果没有激活函数，无论网络多么复杂，最后的输出都是输入的线性组合，而纯粹的线性组合并不能解决更为复杂的问题。
- 引入激活函数之后，由于激活函数都是非线性的，这样就给神经元引入了非线性元素，使得神经网络可以逼近任何非线性函数，这样使得神经网络应用到更多非线性模型中。

激活函数最重要的特性：能求导！

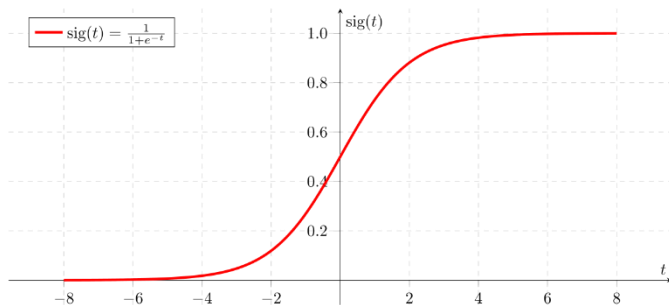
- 常见的且简单的激活函数：

sigmoid

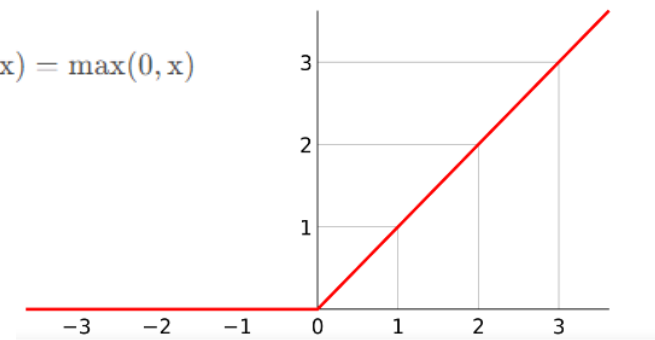
/

relu

$$S(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}$$



$$f(x) = \max(0, x)$$

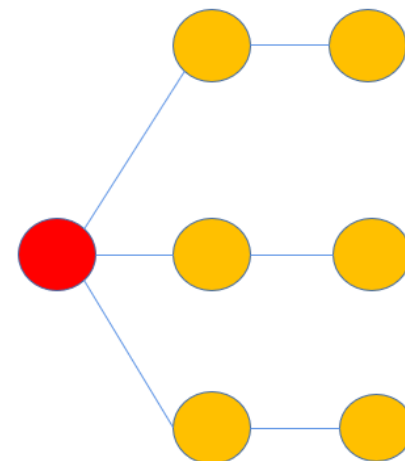
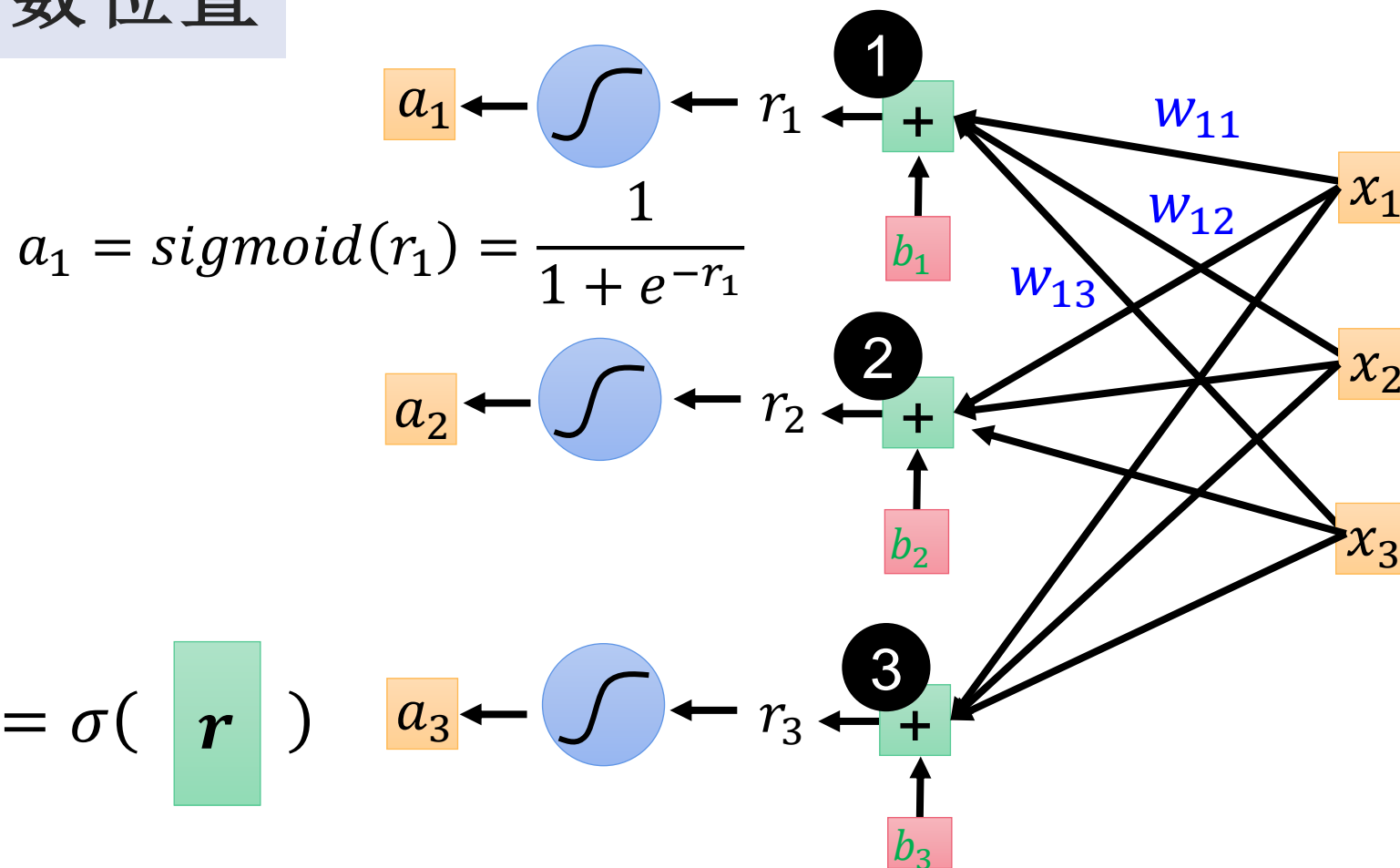


$$\bullet out = \text{sigmoid}(b_1 + w_{11}x_1 + w_{12}x_2 + w_{13}x_3 + w_{14}x_4)$$

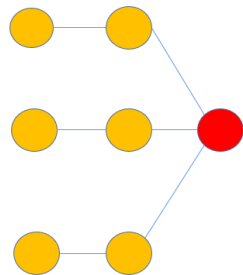


$$\hat{y} = b + \sum_i c_i \text{sigmoid} \left( b_i + \sum_j w_{ij} x_j \right) \quad \begin{matrix} i: 1,2,3 \\ j: 1,2,3 \end{matrix}$$

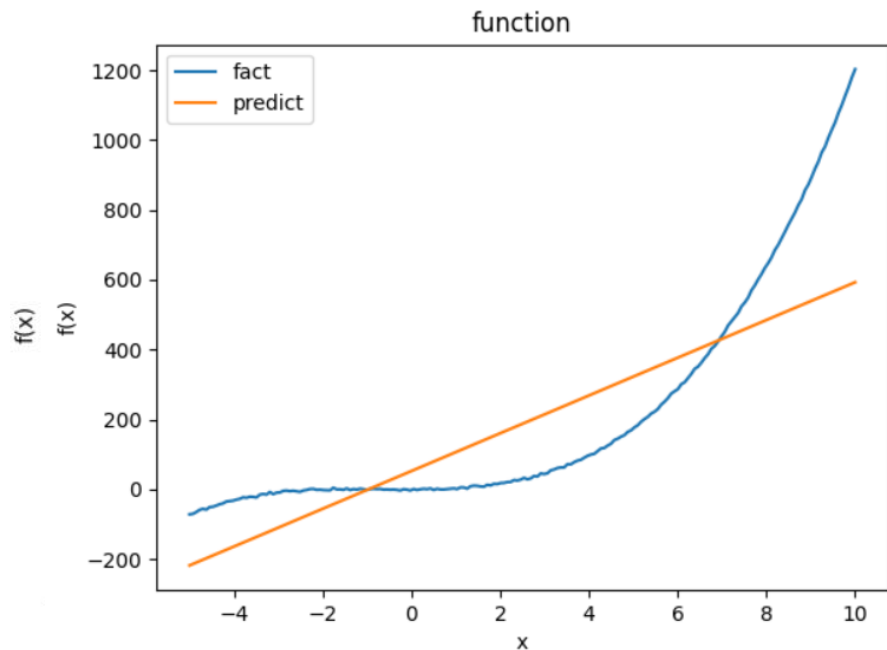
激活函数位置



# 激活函数

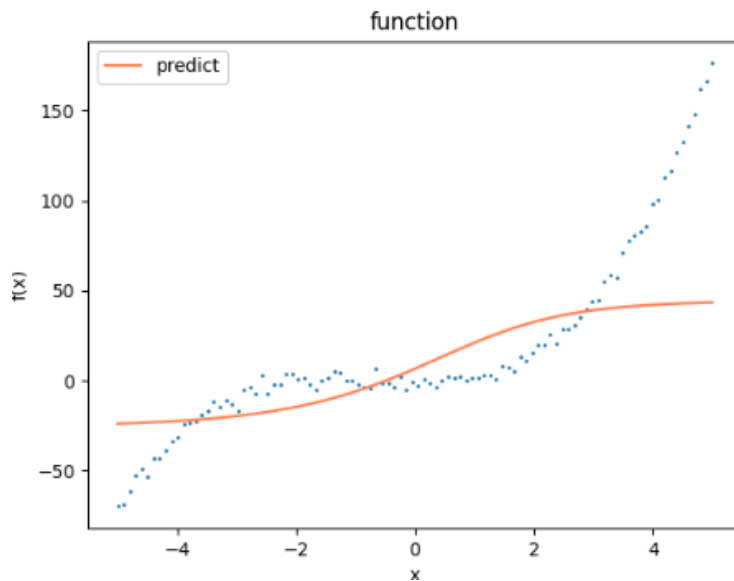


李哥考研

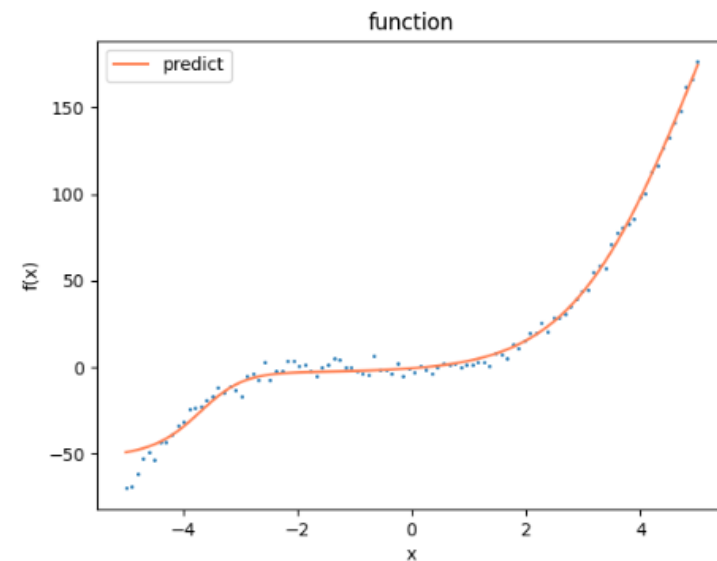


无激活

$$y = x^3 + 2x^2$$



1个sigmoid

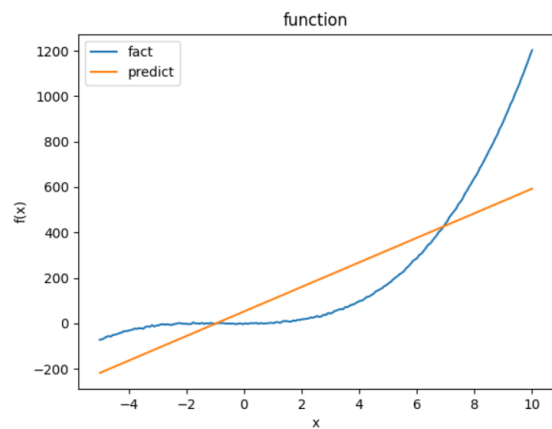


2个sigmoid

# 激活函数

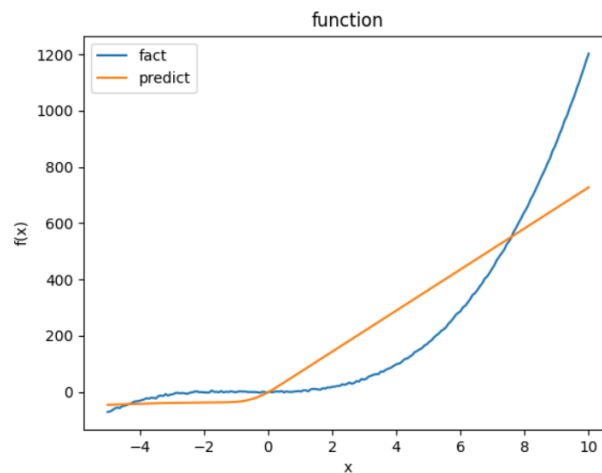


李哥考研

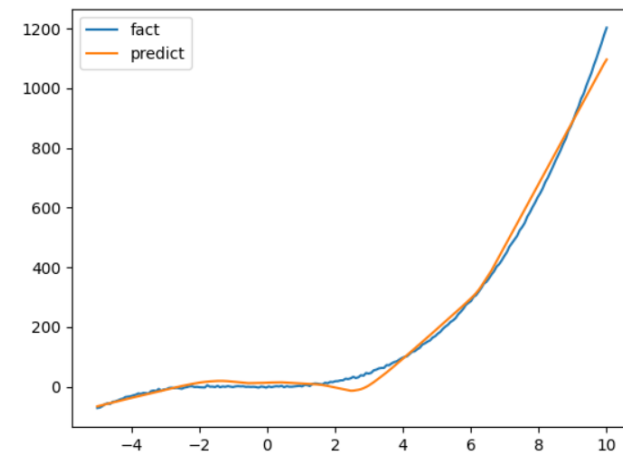


0个relu

$$y = x^3 + 2x^2$$



1个relu

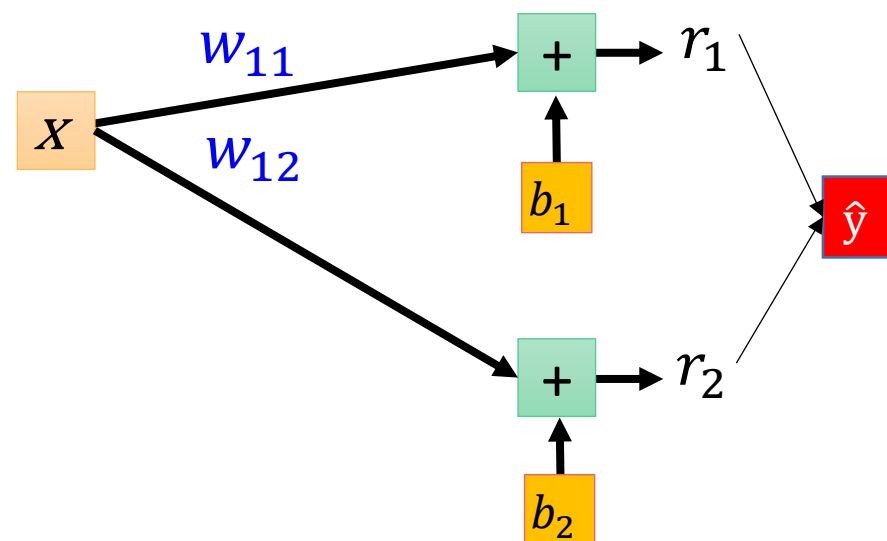
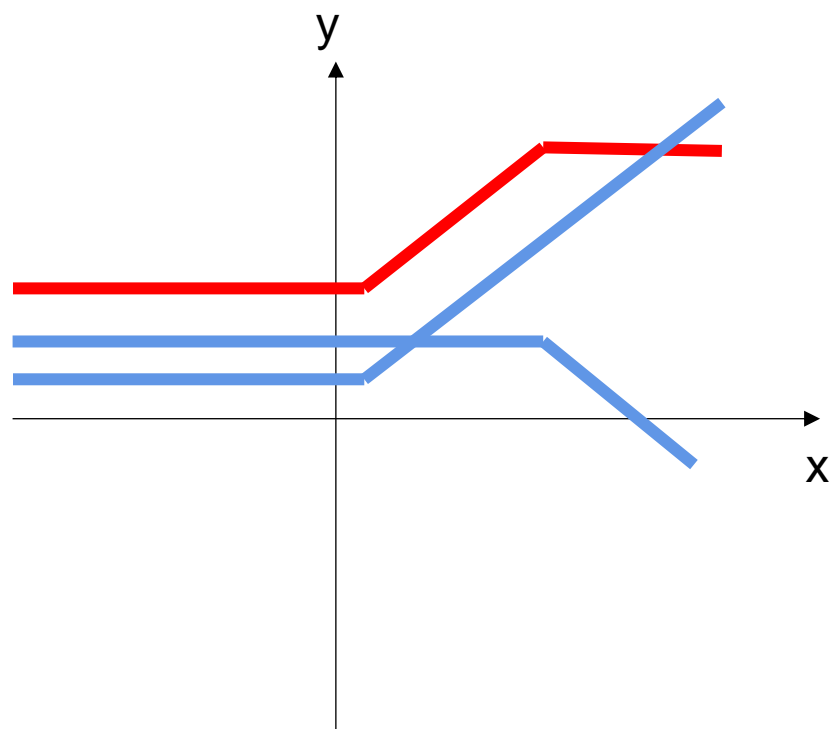


4个relu

# 另一个角度理解拟合和激活函数



李哥考研



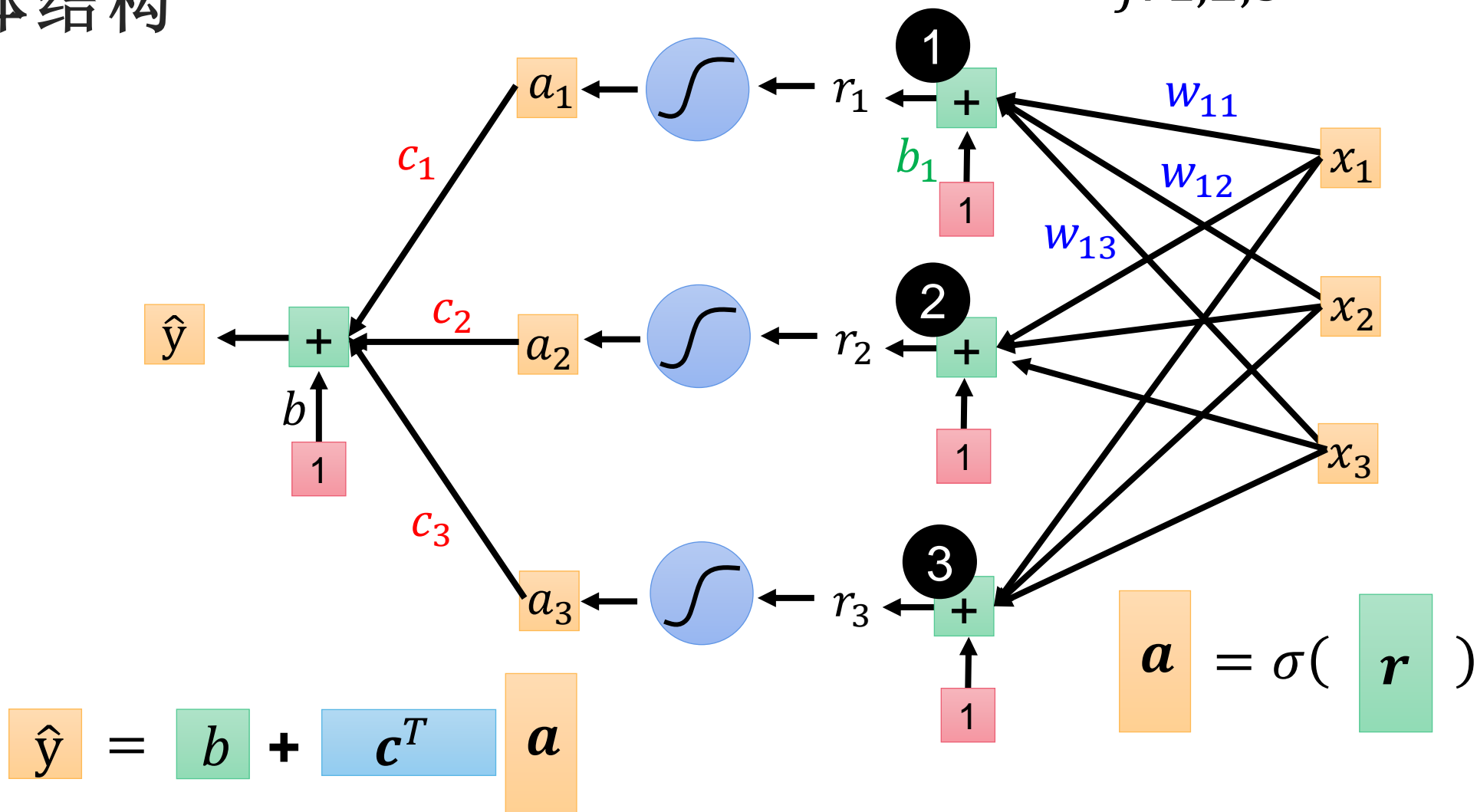


$$y = b + \sum_i \mathbf{c}_i \operatorname{sigmoid} \left( \mathbf{b}_i + \sum_j \mathbf{w}_{ij} x_j \right) \quad \begin{matrix} i: 1,2,3 \\ j: 1,2,3 \end{matrix}$$



李哥考研

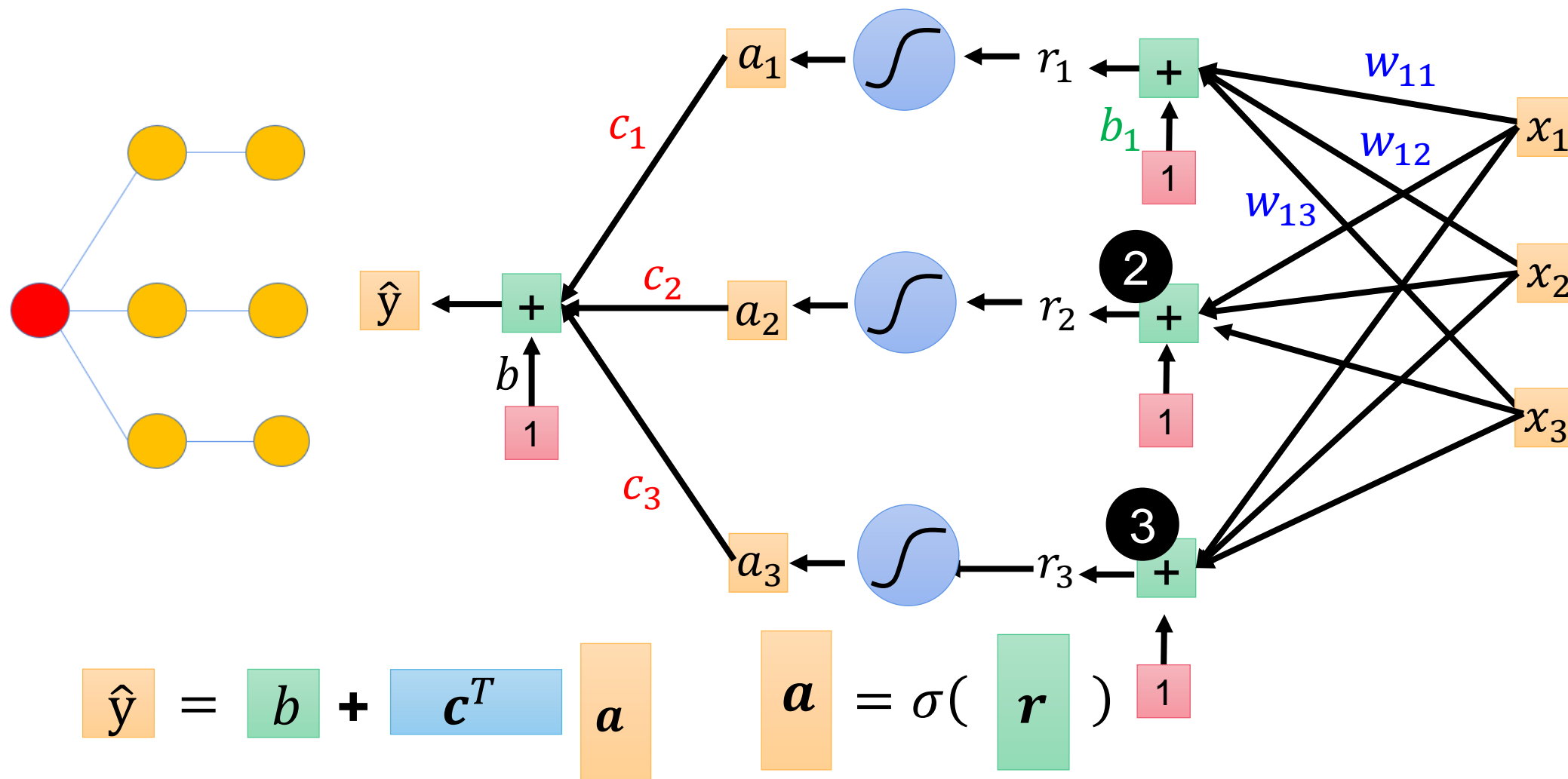
## 模型整体结构



下面三个表示的是同一个东西



李哥考研



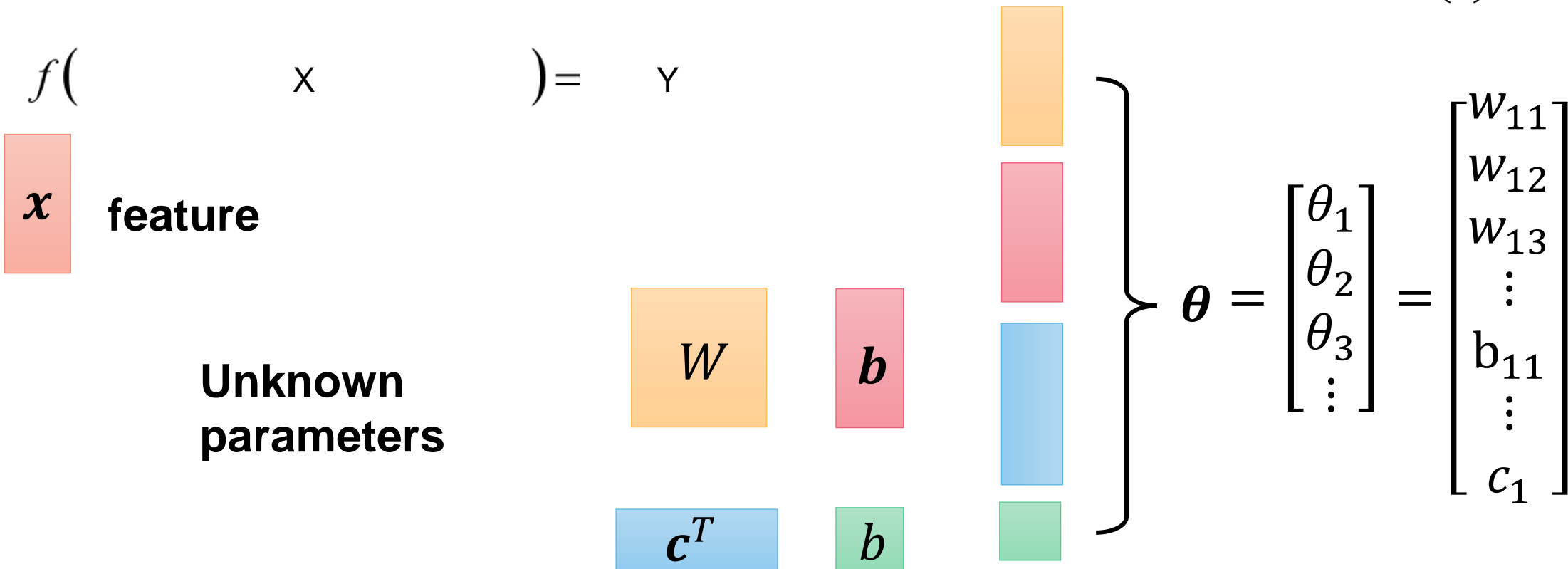
# 神经网络的参数（可训练的）



李哥考研

$$\hat{y} = b + c^T \sigma(b + Wx)$$

$$loss = L(\theta)$$

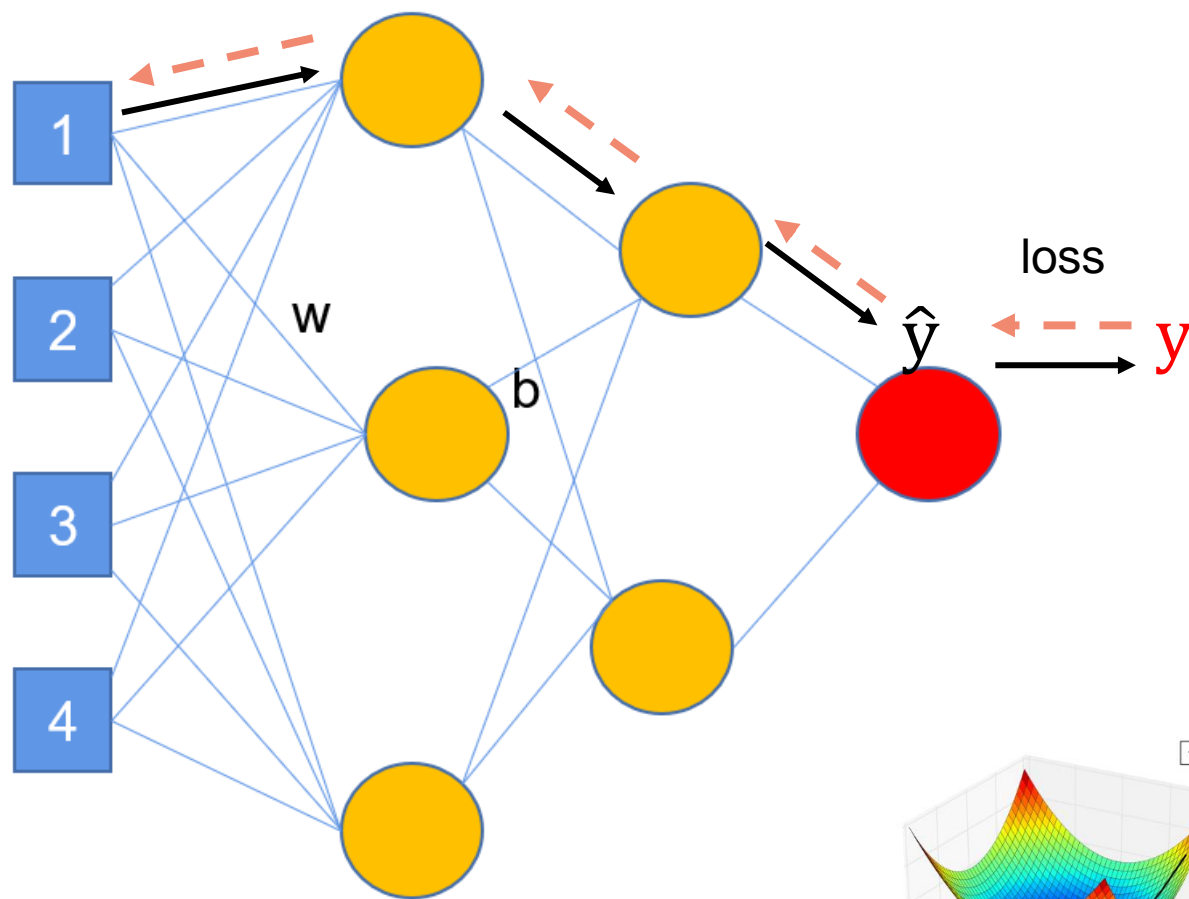


# 深度学习的训练过程。

Step 1:  
定义一个函数  
(模型)

Step 2: 定义一个合  
适的损失函数

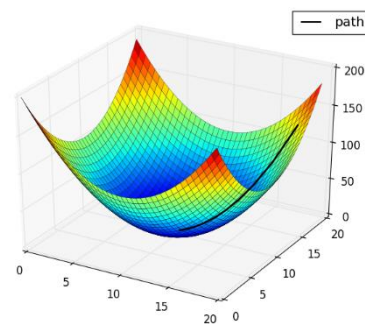
Step 3: 根据损失,  
对模型进行优化



——— : 前向过程

- - - : 梯度回传过程

更新参数  $\theta^0 \rightarrow \theta^1$



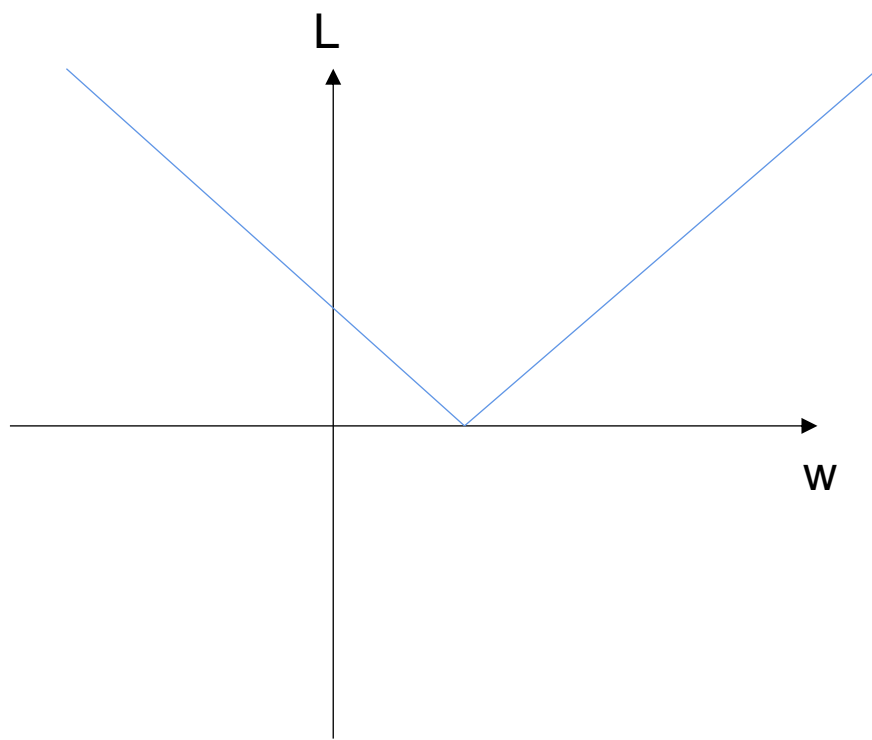
**Gradient**  
**Descent**

此方法称为  
梯度下降

$$\hat{y} = b + c^T \sigma(b + Wx)$$



- $\hat{y} = wx + b \rightarrow \hat{y} = xw + b$        $LOSS = |\hat{y} - y| = |xw + b - y|$



## Gradient Descent

$$w^1 = w^0 - \eta \frac{\partial L}{\partial w}$$

$$b^1 = b^0 - \eta \frac{\partial L}{\partial b}$$

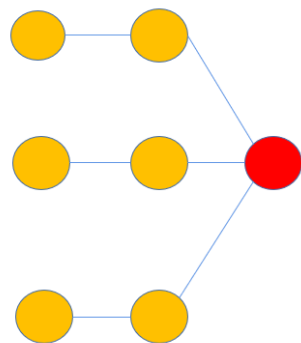
$\eta$ : learning rate

事情不会这么简单。

## Gradient Descent



李哥考研



$$\hat{y} = \sigma( wx + b )$$

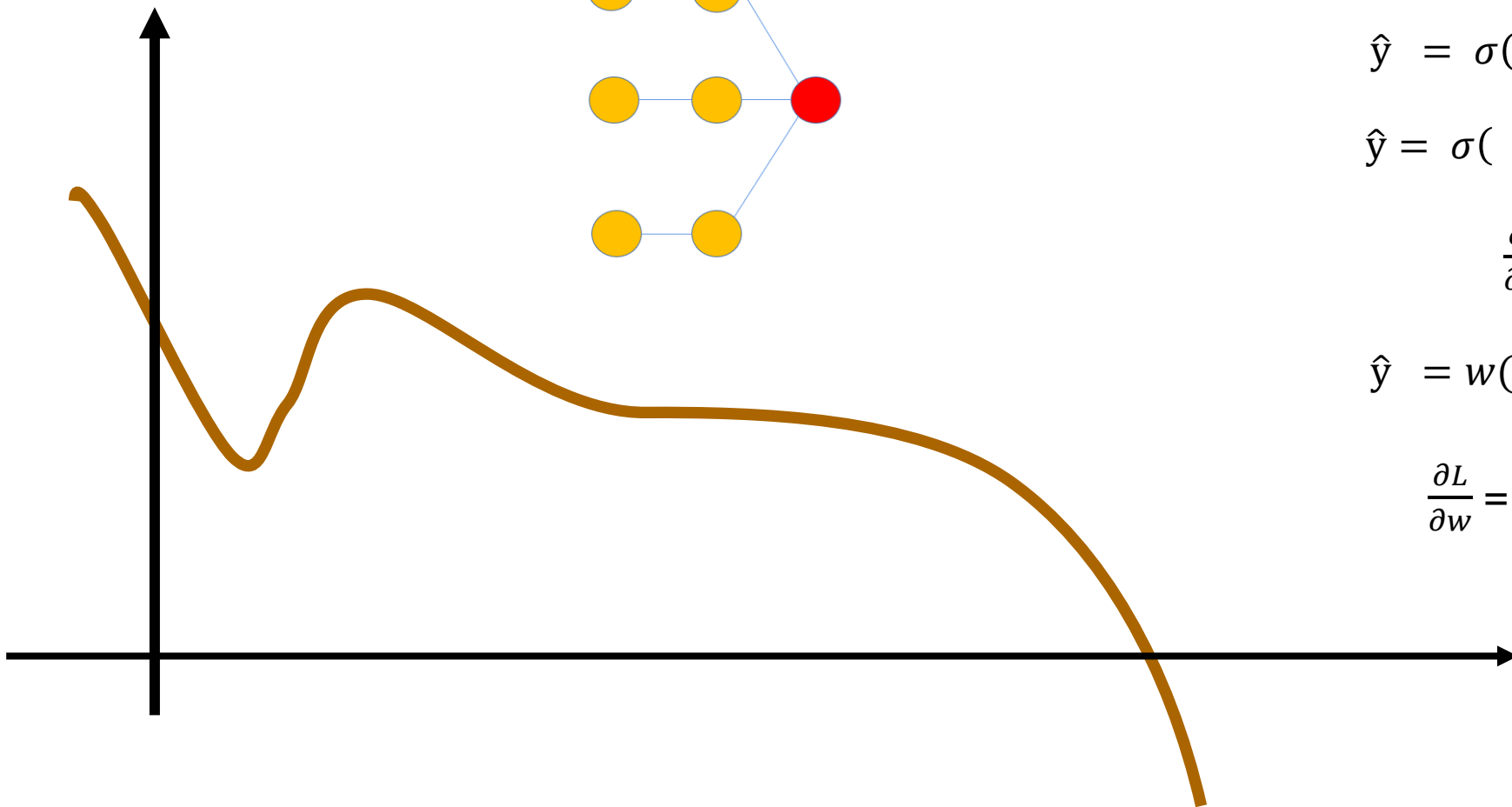
$$\hat{y} = \sigma( r ) \quad r = wx + b$$

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial r} * \frac{\partial r}{\partial w}$$

链式求导。

$$\hat{y} = w(w(...((\sigma( wx + b )...))))+b$$

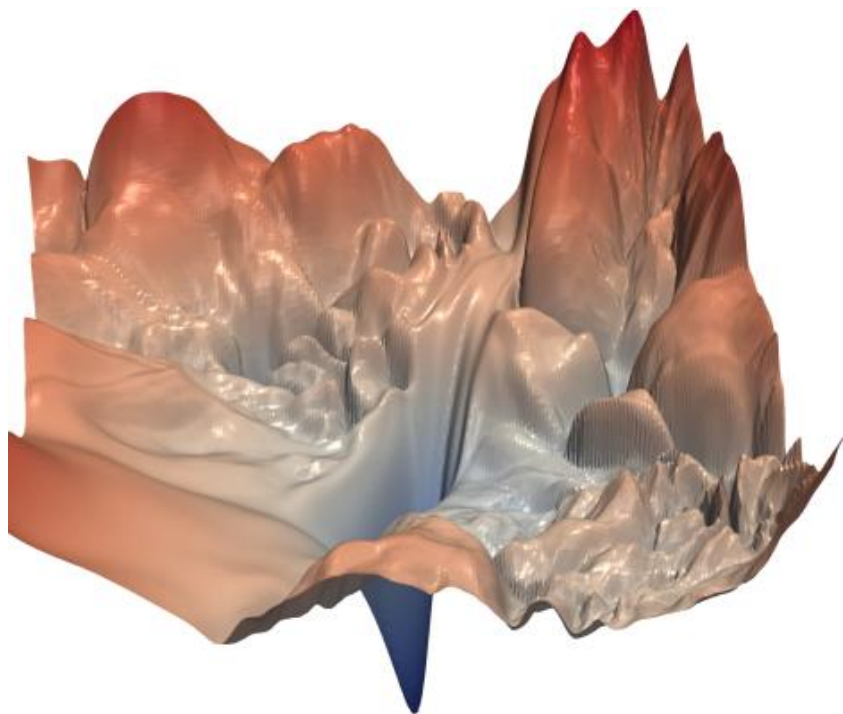
$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial r_n} * \dots * \frac{\partial r_1}{\partial w}$$



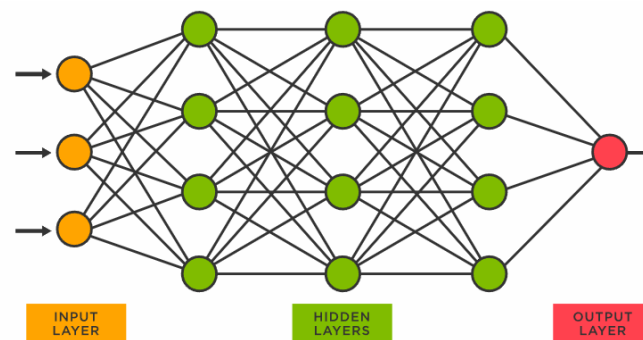
# 事情不会这么简单2



李哥考研



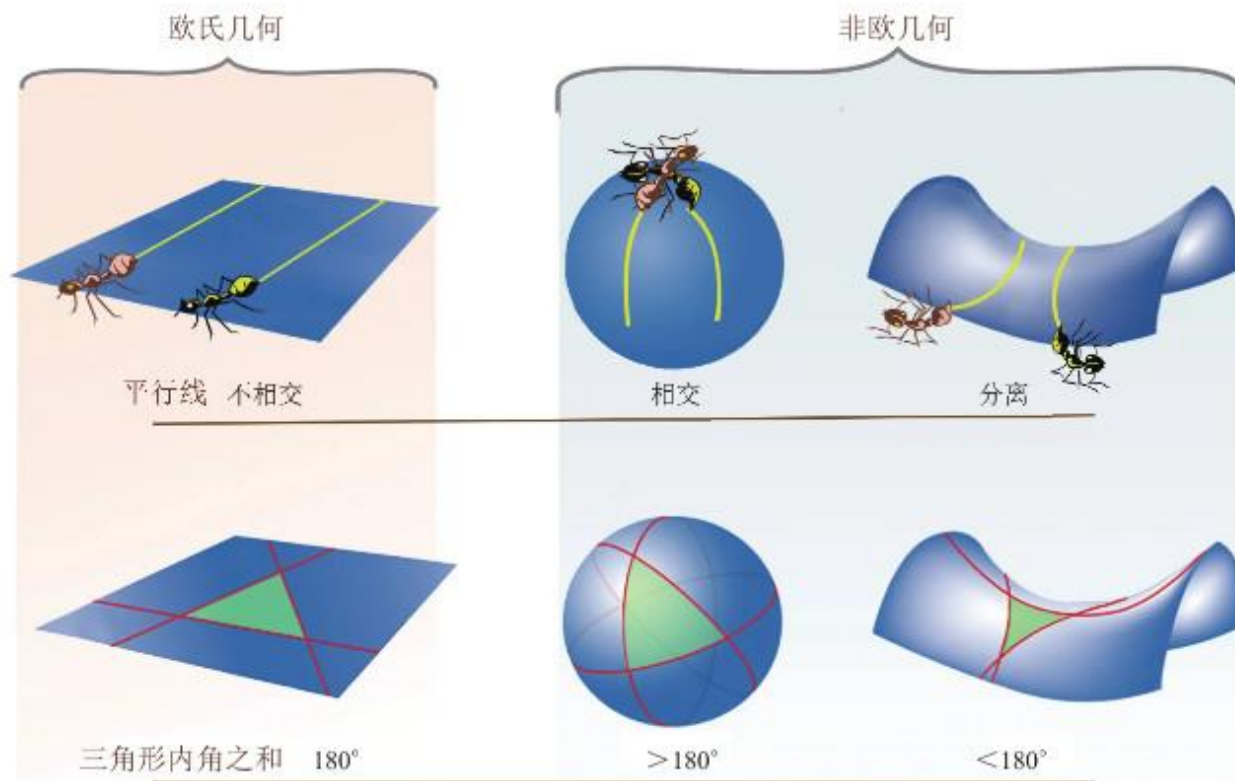
$$\begin{bmatrix} \theta_1^1 \\ \theta_2^1 \\ \vdots \end{bmatrix} \leftarrow \begin{bmatrix} \theta_1^0 \\ \theta_2^0 \\ \vdots \end{bmatrix} - \begin{bmatrix} \eta \frac{\partial L}{\partial \theta_1} \big|_{\theta=\theta^0} \\ \eta \frac{\partial L}{\partial \theta_2} \big|_{\theta=\theta^0} \\ \vdots \end{bmatrix}$$



# 事情不会这么简单3



李哥考研



无所谓，  
我会出手

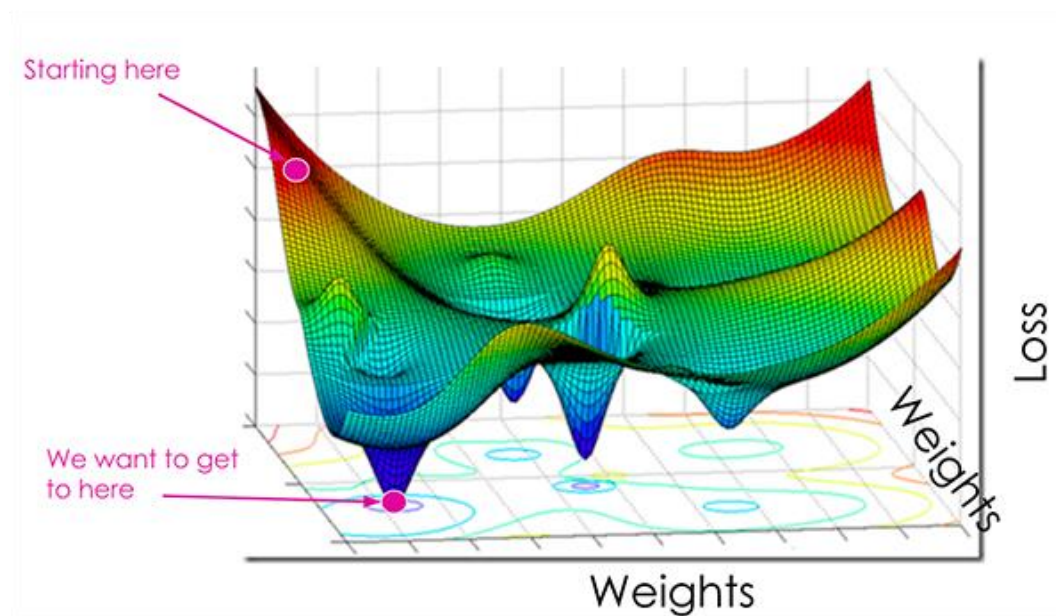
$$\begin{bmatrix} \theta_1^1 \\ \theta_2^1 \\ \vdots \end{bmatrix} \leftarrow \begin{bmatrix} \theta_1^0 \\ \theta_2^0 \\ \vdots \end{bmatrix} - \begin{bmatrix} \eta \frac{\partial L}{\partial \theta_1} |_{\theta=\theta^0} \\ \eta \frac{\partial L}{\partial \theta_2} |_{\theta=\theta^0} \\ \vdots \end{bmatrix}$$



反正最后是得到了一个结果



李哥考研



# 来起个名字吧

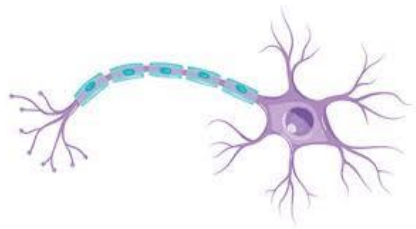


李哥考研

## Fully Connected Network

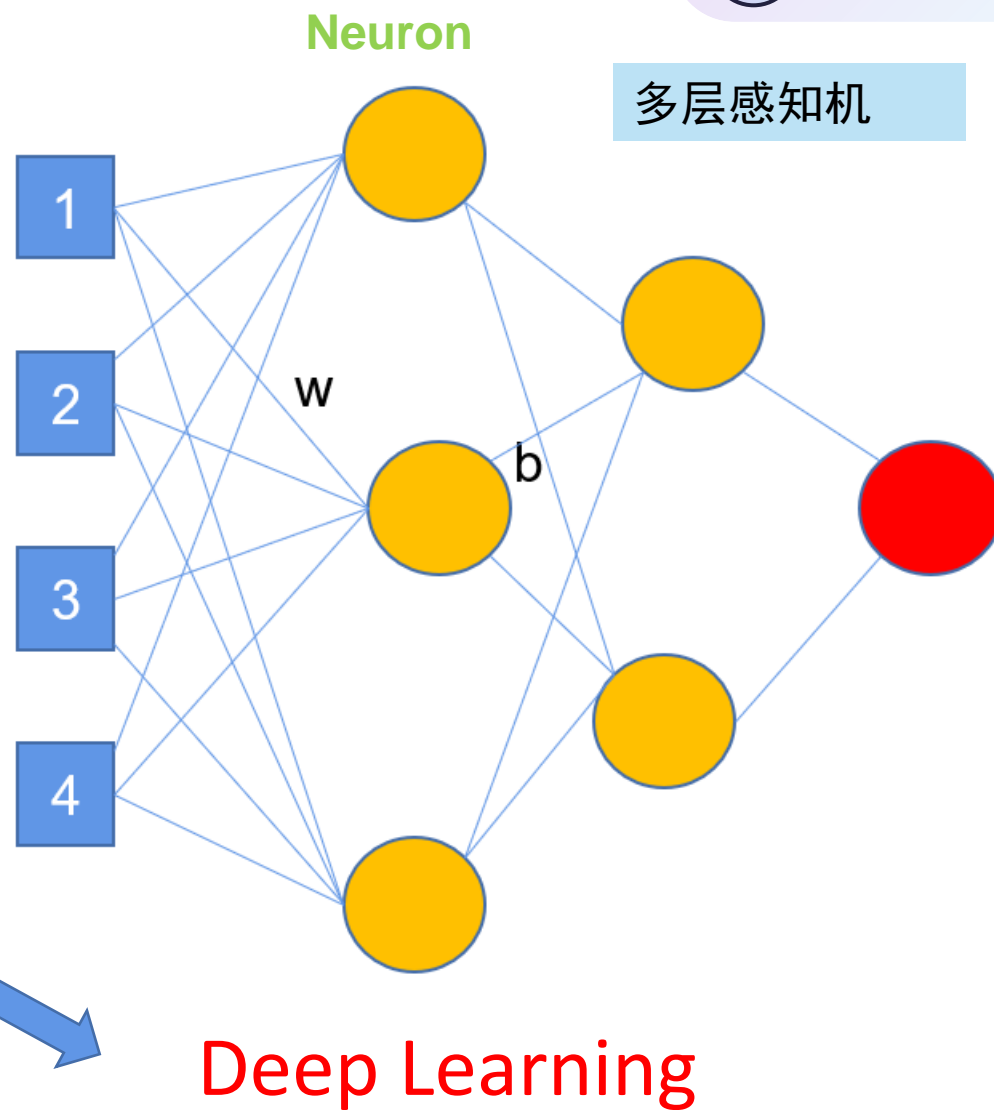
weight

bias



Neural Network

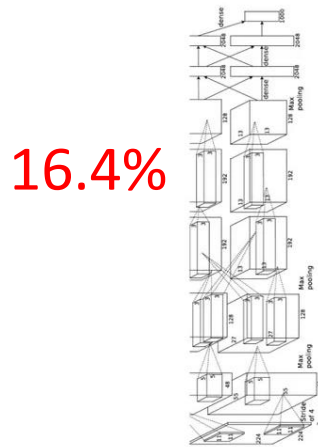
发论文就是在起名字  
一个好的名字是论文的一大半



# 到底有多深啊？

Deep = Many hidden layers

8 layers



AlexNet (2012)

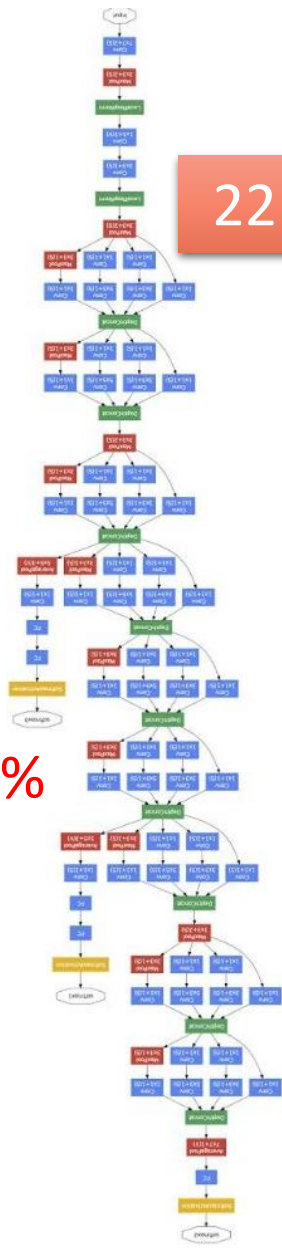
7.3%



VGG (2014)

19 layers

6.7%



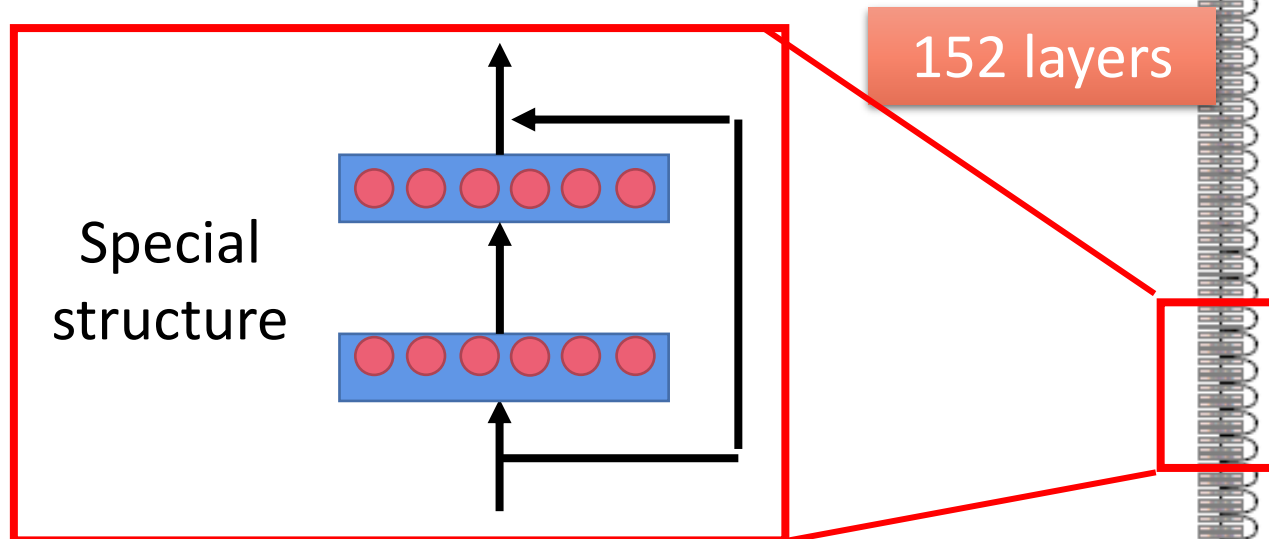
GoogleNet (2014)

22 layers

# Deep = Many hidden layers



李哥考研



101 layers

468米

3.57%

16.4%

AlexNet  
(2012)

7.3%

VGG  
(2014)

6.7%

GoogleNet  
(2014)

Residual Net  
(2015)

成都绿地中心

# 请举一个数学公式的例子。



李哥考研

代码演示时间。

$$y = x^3 + 2x^2$$

1 去掉relu 对非线性函数的表示。

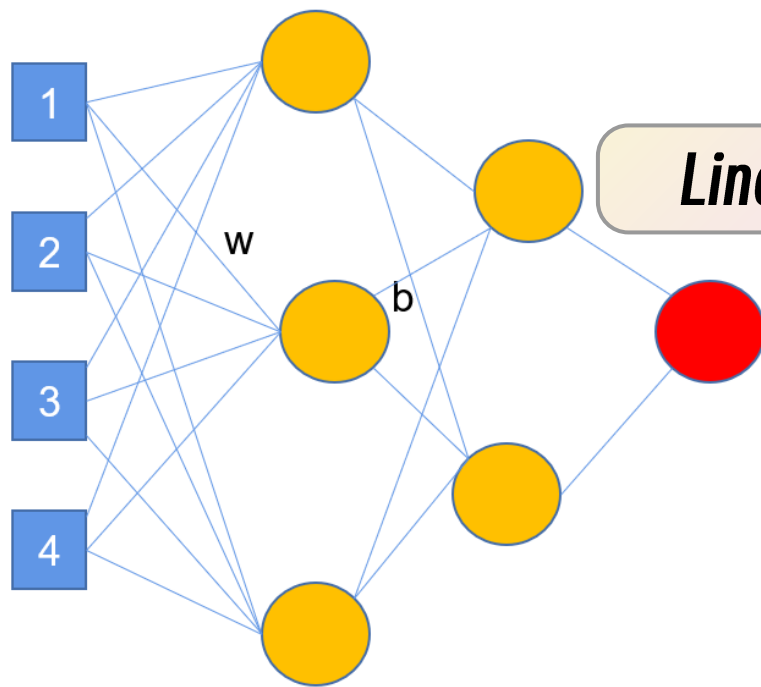
2 浅深层对函数的表示能力。

3 界外预测。

*Linear (4, 3)*

*Linear (3, 2)*

*Linear (2, 1)*



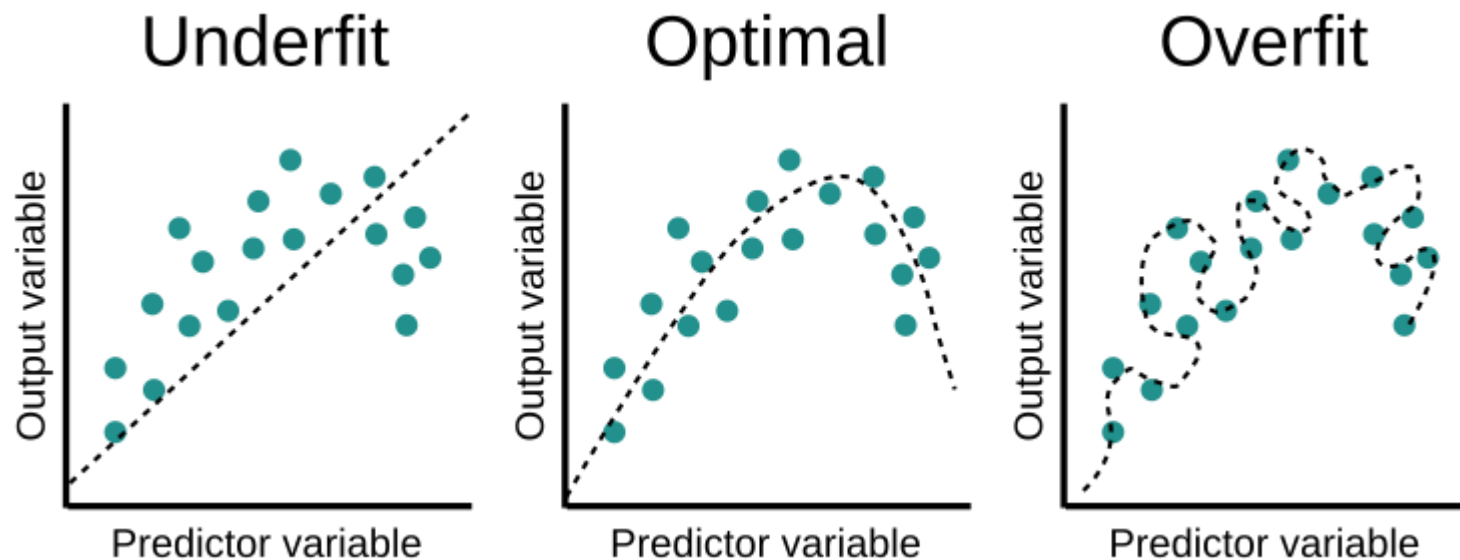
```
nn.Linear( in_features=1, out_features=100)
```

# 过拟合与欠拟合。



李哥考研

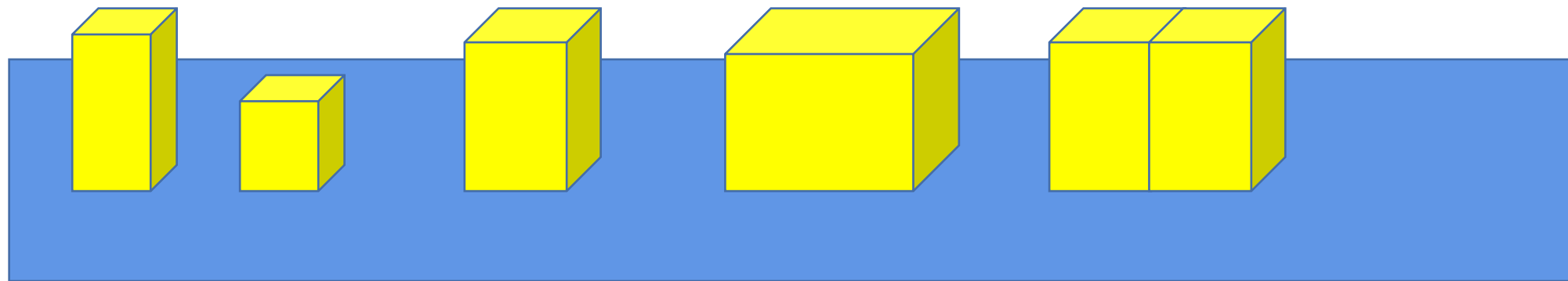
- 过拟合（overfitting）和欠拟合（underfitting）



# 神经网络透析1：架构设计



李哥考研



函数

函数的结构 = 模型的架构

个人经验（玄学）

# 神经网络透析2



李哥考研

- 神经网络， 可以完成超级复杂的任务。诸如图片生成， 人脸识别等等。

- 但回归到一些原始纯粹的简单问题上， 它表现得可能没那么好。

如： 刚才的模拟函数

判断一个数字是否为偶数。

发现数字规律



先验知识





课堂作业： 环境配置， python 基本语法学习

B站搜小甲鱼 学一些基本的python操作。



李哥考研

下节预告。

# Python入门

- 一门语言。

1, 陈述 : `a = b`

2, 循环 : `for i in range(10):`  
`print(i)`

3, 判断 : `if a > 3:`  
`print(a)`

4, 函数 `def func(x):`  
`return x`

data structure

circulation

judge

function

class

assignment

random

numpy

tensor

切片

包的引用

矩阵

张量

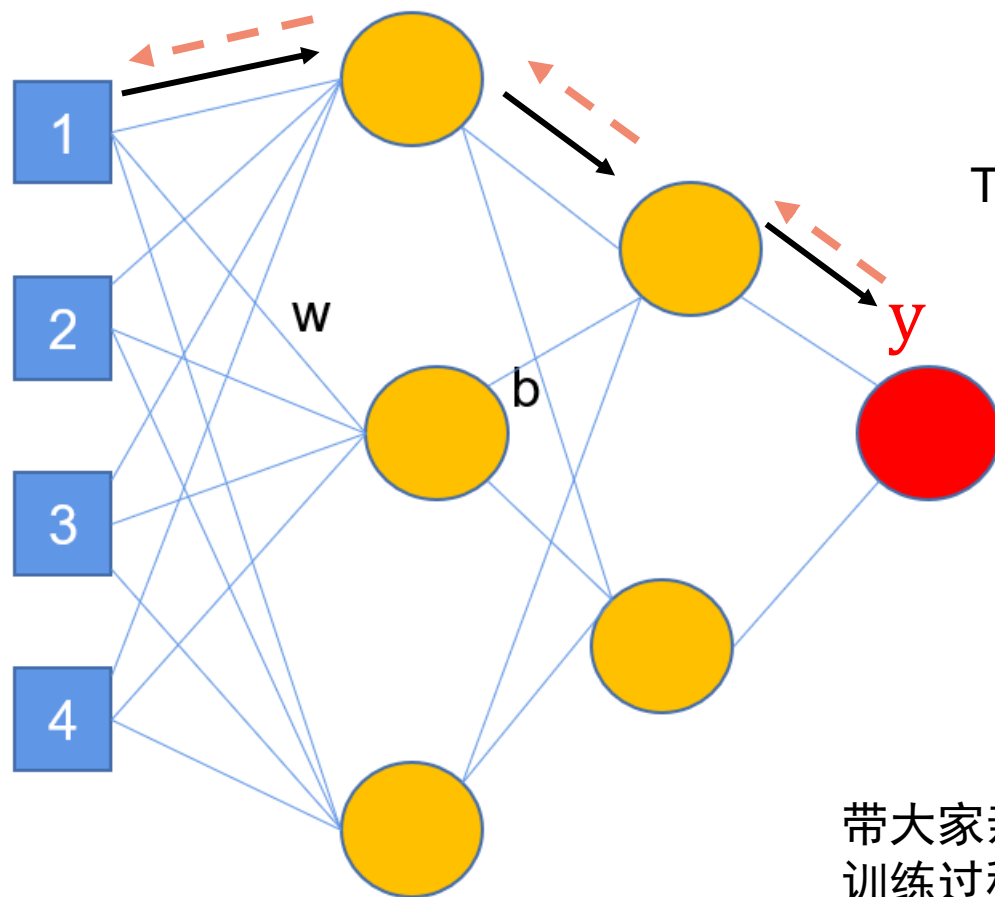


李哥考研

# Pytorch



李哥考研



Tensor张量网。

Detach 表示取下来

带大家亲手从0 写一个线性回归。知道一个神经网络基本训练过程。



李哥考研

# 下下节预告。

1, 实战， 新冠人数预测。



李哥考研

答疑

THANKS