

第十部分：误差反向传播算法（BP）

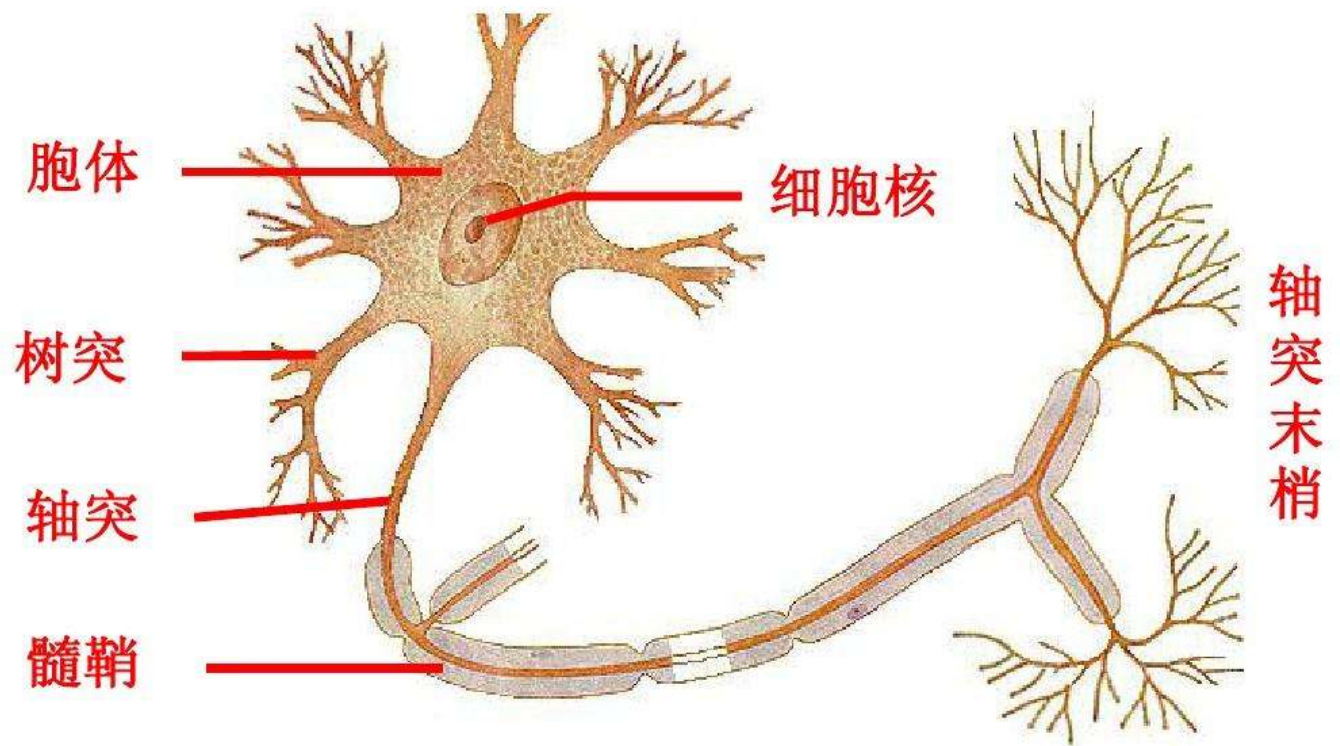
- 第十部分：误差反向传播算法（BP）
 - 一、神经网络基本概念
 - 二、神经元模型
 - 1. 单个神经元
 - 1> 感知器模型
 - 2> 输入与输出
 - 3> 激活函数
 - 2. 单个神经元的学习
 - 1> 识别
 - 3. 多层神经元模型
 - 三、误差反向传播算法(BP)
 - 1. 算法的推导
 - 1> 输出层权重偏导
 - 2> 隐藏层权重偏导
 - 3. 训练
 - 1> 调整单个权重 w 。我们追求误差减小，故，有以下公式：
 - 2> 随机梯度下降
 - 4. 编码
 - 四、一个学习示例
 - 1. 性别预测
 - 五、算法优化
 - 1. 指定网络结构
 - 2. 存储当前神经网络的训练数据

一、神经网络基本概念

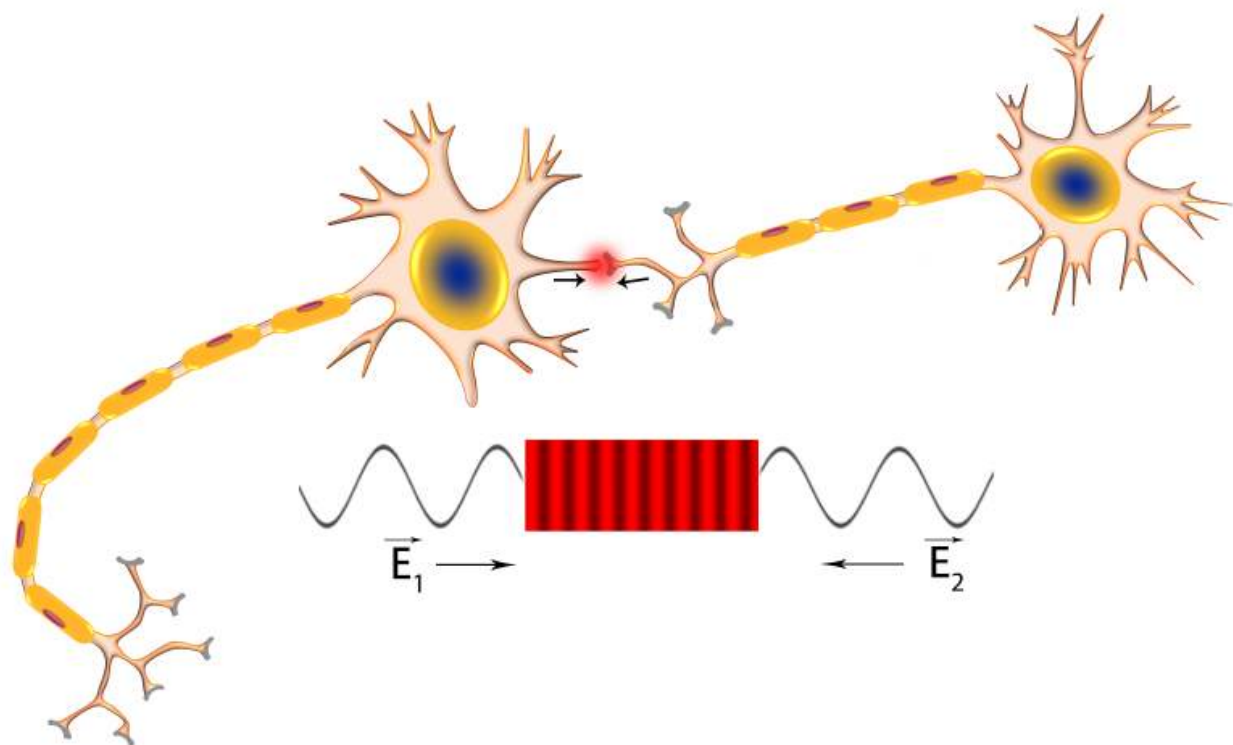
二、神经元模型

神经元细胞示意图

二、神经元的结构

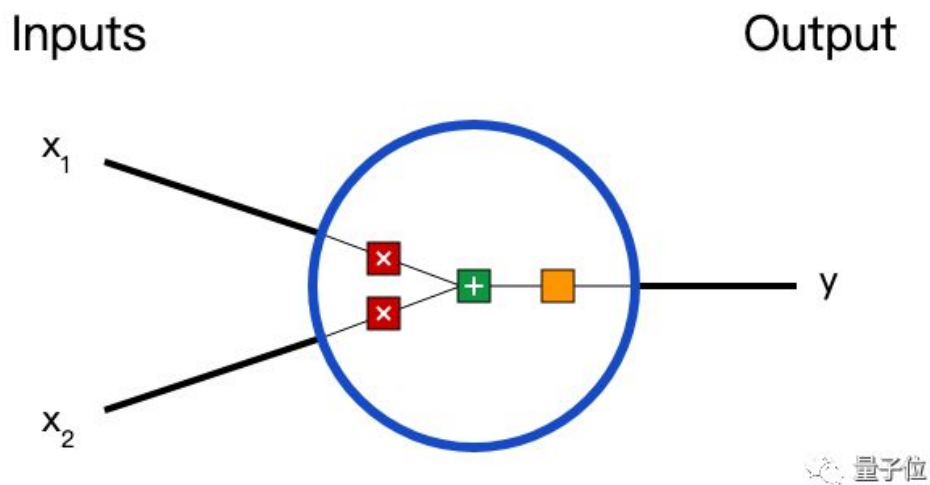


髓鞘是包裹在神经细胞轴突外面的一层膜



1. 单个神经元

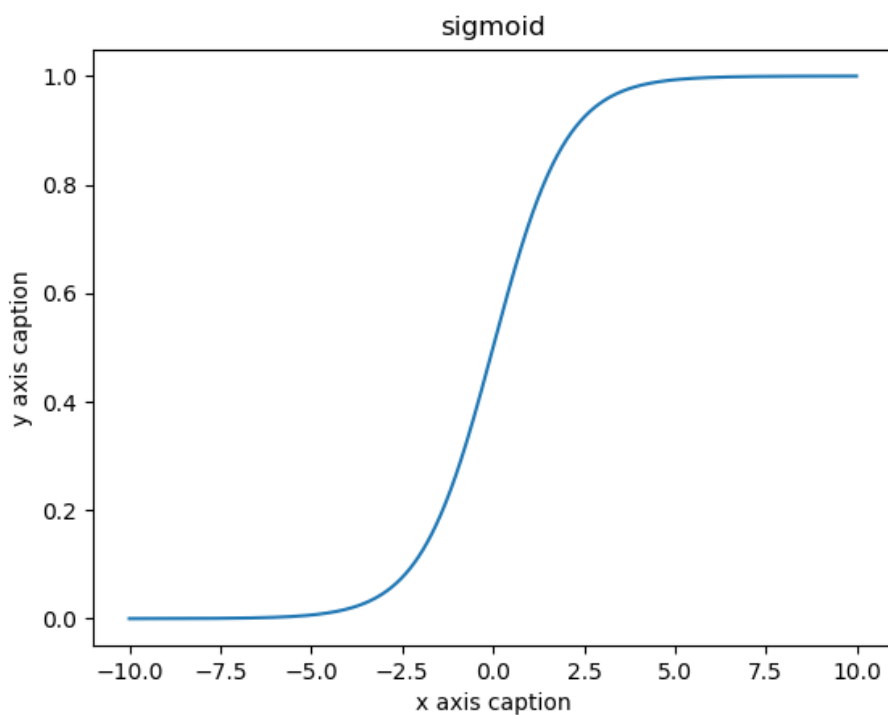
1> 感知器模型



2> 输入与输出

3> 激活函数

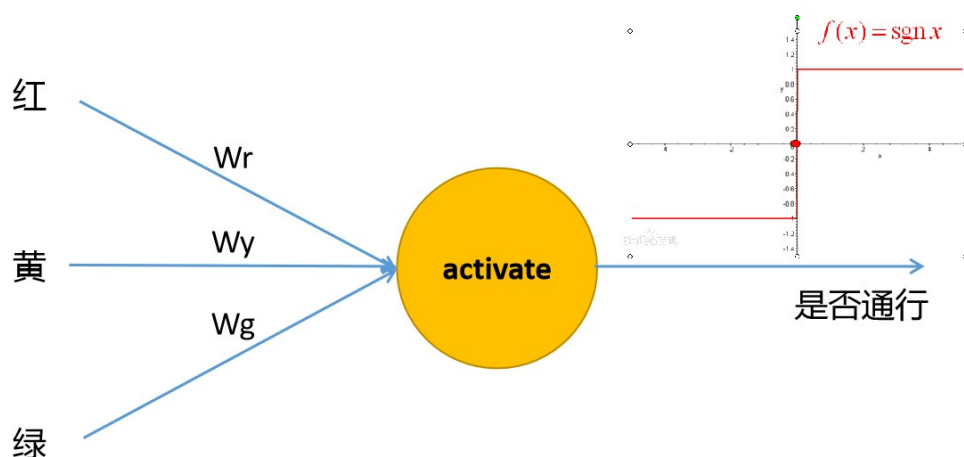
$$Sigmoid(x) = \frac{1}{1 + e^{-x}}$$



$$Sigmoid'(x) = Sigmoid(x) * (1 - Sigmoid(x))$$

2. 单个神经元的学习

1> 识别



$$Y = \text{activate}(X_r * W_r + X_y * W_y + X_g * W_g)$$

拟合函数：

选择激活函数使用符号函数： $\text{sign}(x)$

X1:[1, 0, 0] --> Y:-1

X2:[0, 1, 0] --> Y:-1

X3:[0, 0, 1] --> Y:1

$$Y = \text{sign}(X_r * W_r + X_y * W_y + X_g * W_g)$$

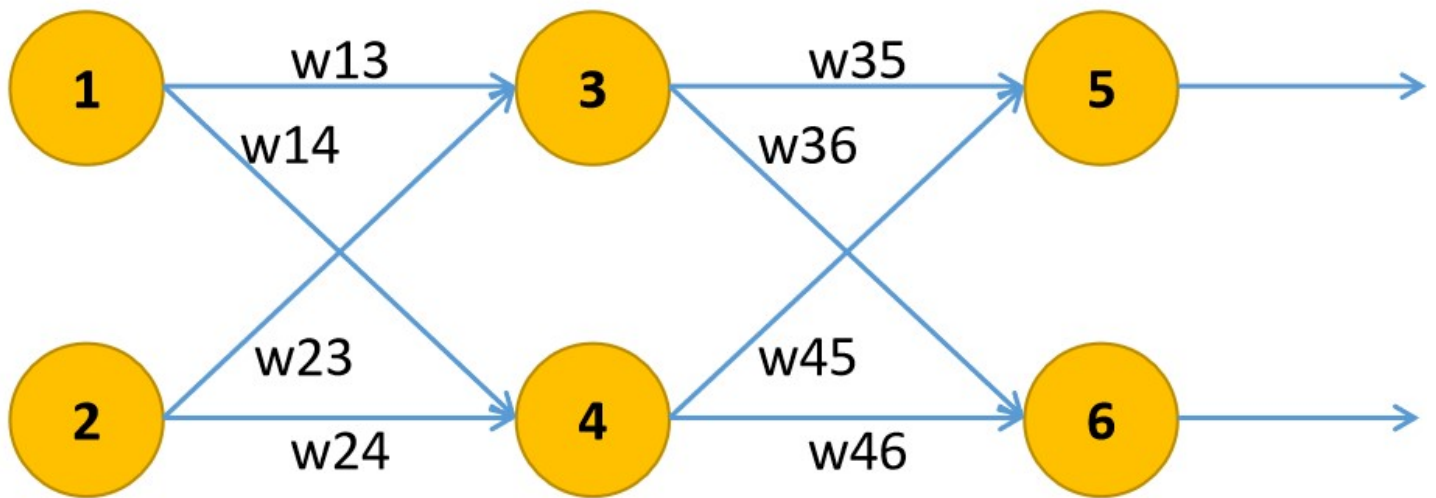
识别交通信号灯的一个解： $W = [-1, -1, 1]$

3. 多层神经元模型

输入层

隐藏层

输出层

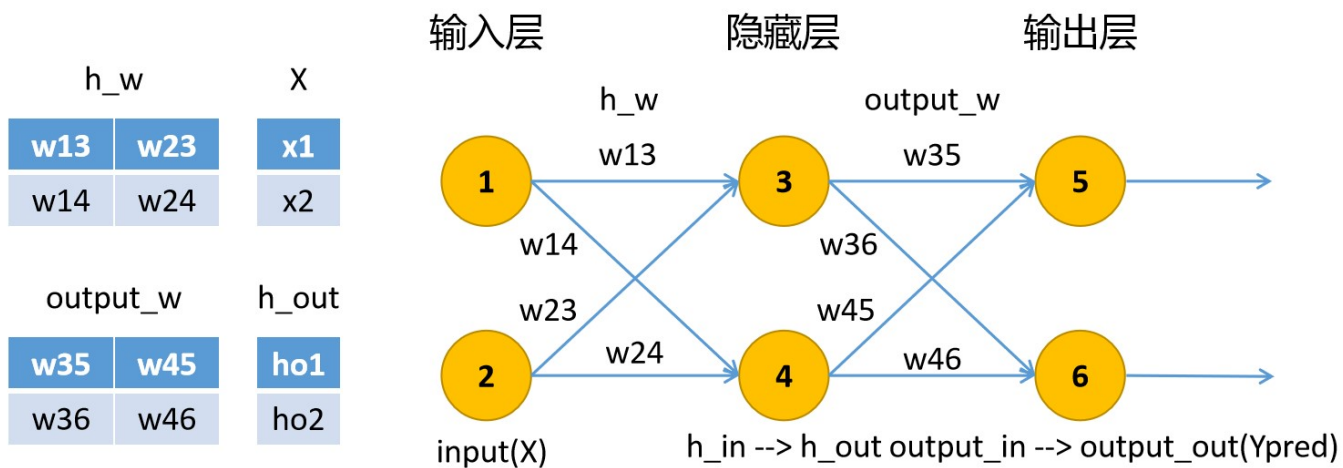


input

$h_{in} \rightarrow h_{out}$ $output_{in} \rightarrow output_{out}$

三、误差反向传播算法(BP)

1. 算法的推导



$h_in = h_w \times X$
 $h_out = \text{Sigmoid}(h_in)$
 $output_in = output_w \times h_out$
 $output_out = \text{Sigmoid}(output_in)$

$Error = D(Y) = 1/2 * \sum (Y - Y_{pre})^2 = F(X, w_{ij}, Y_{true})$

input --[h_w]--> h_in --[sigmoid]--> h_out --[output_w]-->output_in --[sigmoid]--> output_out-->Error

符号说明：

- x：输入向量
- Y_{true} : 输入向量 x 对应的真实结果
- Y_{pred} : 根据输入向量 x 预测的结果
- y_i ：向量的某个分量
- h_in：隐藏层输入
- h_out：隐藏层输出(经过激活函数处理)
- h_w：隐藏层权重矩阵
- output_in：输出层输入
- output_out：输出层输出(经过激活函数处理)
- output_w：输出层权重矩阵

1> 输出层权重偏导

已知对于某个训练样例 d 的误差函数：

$$Error = \frac{1}{2} * \sum_{i \in output} (Y_{i_{true}} - Y_{i_{pred}})^2$$

可得，总误差函数对输出层权重 w_{35} 的偏导数：

$$\frac{\partial E}{\partial w_{35}} = \frac{\partial E}{\partial output5_out} * \frac{\partial output5_out}{\partial output5_in} * \frac{\partial output5_in}{\partial w_{35}}$$

各项展开得：

$$\begin{aligned}\frac{\partial E}{\partial output5_out} &= -(Y5_{true} - output5_out) \\ \frac{\partial output5_out}{\partial output5_in} &= Sigmoid(output5_in) * (1 - Sigmoid(output5_in)) \\ output5_in &= h3_out * w_{35} + h4_out * w_{45} \\ \frac{\partial output_in}{\partial w_{35}} &= h3_out\end{aligned}$$

最终偏导为：

$$\frac{\partial E}{\partial w_{35}} = -(Y5_{true} - output5_out) * Sigmoid(output5_in) * (1 - Sigmoid(output5_in)) * h3_out$$

2> 隐藏层权重偏导

总体上，隐藏层权重偏导公式如下：

$$\frac{\partial E}{\partial w_{13}} = \frac{\partial E}{\partial h3_out} * \frac{\partial h3_out}{\partial h3_in} * \frac{\partial h3_in}{\partial w_{13}}$$

展开，得：

$$\frac{\partial E}{\partial w_{13}} = \frac{\partial E}{\partial h3_out} * Sigmoid(h3_in) * (1 - Sigmoid(h3_in)) * x_1$$

仔细分析，我们只需对权重 w_{13} 直接影响的节点求导即可：

$$\frac{\partial E}{\partial h3_out} = \sum_{i \in Downstream(h3)} \frac{\partial E}{\partial outputi_out} * \frac{\partial outputi_out}{\partial outputi_in} * \frac{\partial outputi_in}{\partial h3_out}$$

各项展开得：

$$\frac{\partial E}{\partial h3_out} = \sum_{i \in Downstream(h3)} -(Yi_{true} - outputi_out) * Sigmoid(outputi_in) * (1 - Sigmoid(outputi_in)) * w_{3i}$$

3. 训练

1> 调整单个权重 w 。我们追求误差减小，故，有以下公式：

$$w_{35} = w_{35} - \Delta w_{35}$$

定义学习速率： r

$$w_{35} = w_{35} - r * \frac{\partial E}{\partial w_{35}}$$

输出层权重调整：

$$w_{35} = w_{35} + r * (Y5_{true} - output5_out) * Sigmoid(output5_in) * (1 - Sigmoid(output5_in)) * h3_out$$

$$w_{45} = w_{45} + r * (Y5_{true} - output5_out) * Sigmoid(output5_in) * (1 - Sigmoid(output5_in)) * h4_out$$

$$w_{36} = w_{36} + r * (Y_{6_{true}} - output_{6_out}) * Sigmoid(output_{6_in}) * (1 - Sigmoid(output_{6_in})) * h_{3_out}$$

$$w_{46} = w_{46} + r * (Y_{6_{true}} - output_{6_out}) * Sigmoid(output_{6_in}) * (1 - Sigmoid(output_{6_in})) * h_{4_out}$$

利用矩阵运算简化

$$hw+ = r * \begin{pmatrix} Y_{5_{true}} - Y_{5_{pred}} \\ Y_{6_{true}} - Y_{6_{pred}} \end{pmatrix} \cdot Sigmoid' \begin{pmatrix} output_{5_in} \\ output_{6_in} \end{pmatrix} \times \begin{pmatrix} h_{3_out} & h_{4_out} \end{pmatrix}$$

继续化简(默认向量为列向量):

$$hw+ = r * (Y_{true} - Y_{pred}) \cdot Sigmoid'(output_in) \times h_out.T$$

令:

$$output_error = (Y_{true} - Y_{pred})$$

则:

$$hw+ = r * (output_errors \cdot Sigmoid'(output_in)) \times h_out.T$$

隐藏层权重调整:

$$w_{13} = w_{13} + r * Sigmoid'(h_{3_in}) * x_1 * \sum_{i \in Downstream(h3)} (Y_{i_{true}} - output_{i_out}) * Sigmoid'(output_{i_in}) * w_{3i}$$

$$w_{23} = w_{23} + r * Sigmoid'(h_{3_in}) * x_2 * \sum_{i \in Downstream(h3)} (Y_{i_{true}} - output_{i_out}) * Sigmoid'(output_{i_in}) * w_{3i}$$

$$w_{14} = w_{14} + r * Sigmoid'(h_{4_in}) * x_1 * \sum_{i \in Downstream(h4)} (Y_{i_{true}} - output_{i_out}) * Sigmoid'(output_{i_in}) * w_{4i}$$

$$w_{24} = w_{24} + r * Sigmoid'(h_{4_in}) * x_2 * \sum_{i \in Downstream(h4)} (Y_{i_{true}} - output_{i_out}) * Sigmoid'(output_{i_in}) * w_{4i}$$

利用矩阵简化:

$$hw_{ij}+ = r * Sigmoid' \begin{pmatrix} h_{j_in} \\ h_{j_in} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \times \sum_{k \in Downstream(h_j)} (Y_{k_{true}} - Y_{k_{pred}}) \cdot Sigmoid'(output_{k_in}) \times w_{jk}$$

令(默认向量为列向量):

$$hidden_errors = output_w.T \times ((Y_{k_{true}} - Y_{k_{pred}}) \cdot Sigmoid'(output_in))$$

带入得(默认向量为列向量):

$$hw+ = r * (hidden_errors \cdot Sigmoid'(h_in)) \times X.T$$

2> 随机梯度下降

1. 对训练样本中的每一个数据，进行一次训练
2. 对整个样本进行多次(例如：100次)训练

4. 编码

四、一个学习示例

1. 性别预测

五、 算法优化

1. 指定网络结构

可随意指定：输入层，隐藏层，输出层的神经元个数

2. 存储当前神经网络的训练数据

将神经网络结构，包括：学习速率，训练次数和所有权重矩阵存成文件，以便以后可以不必训练，加载数据后可直接进行预测。