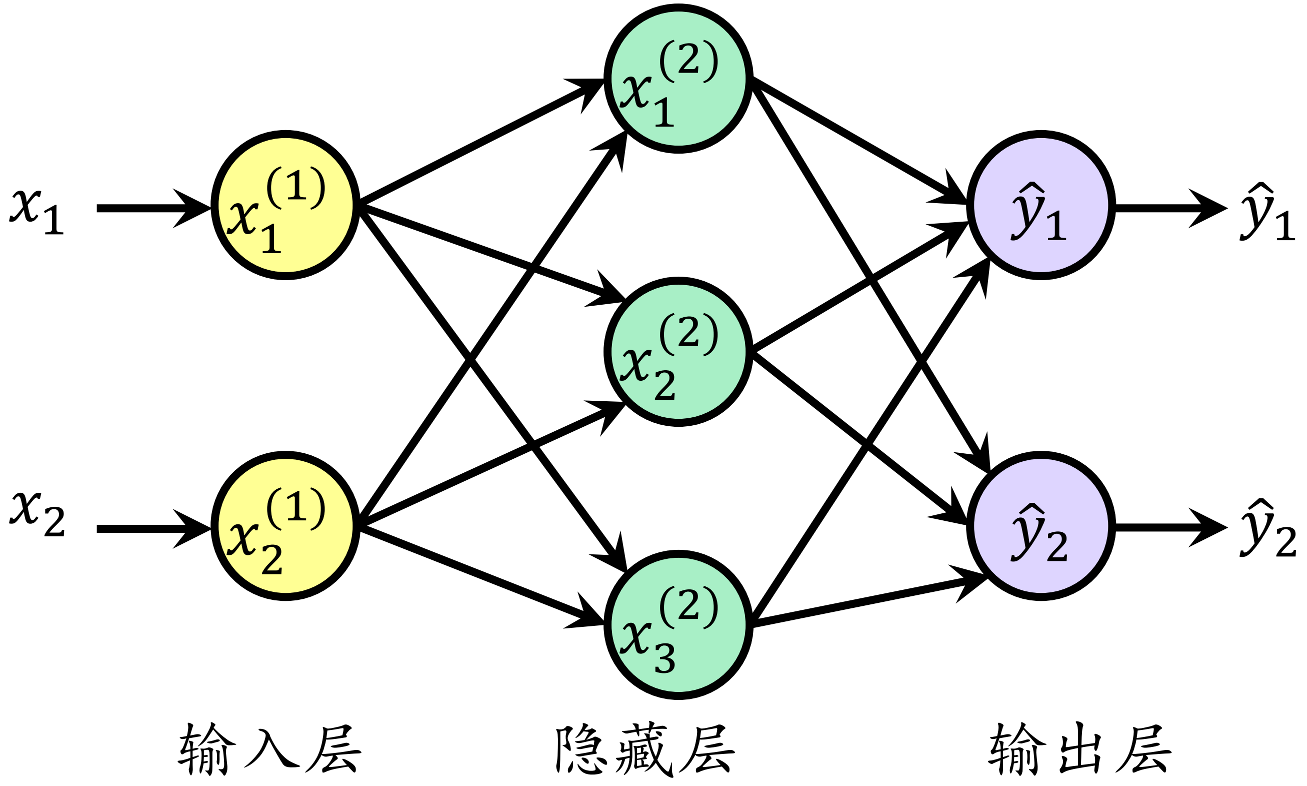
神经网络

李 波

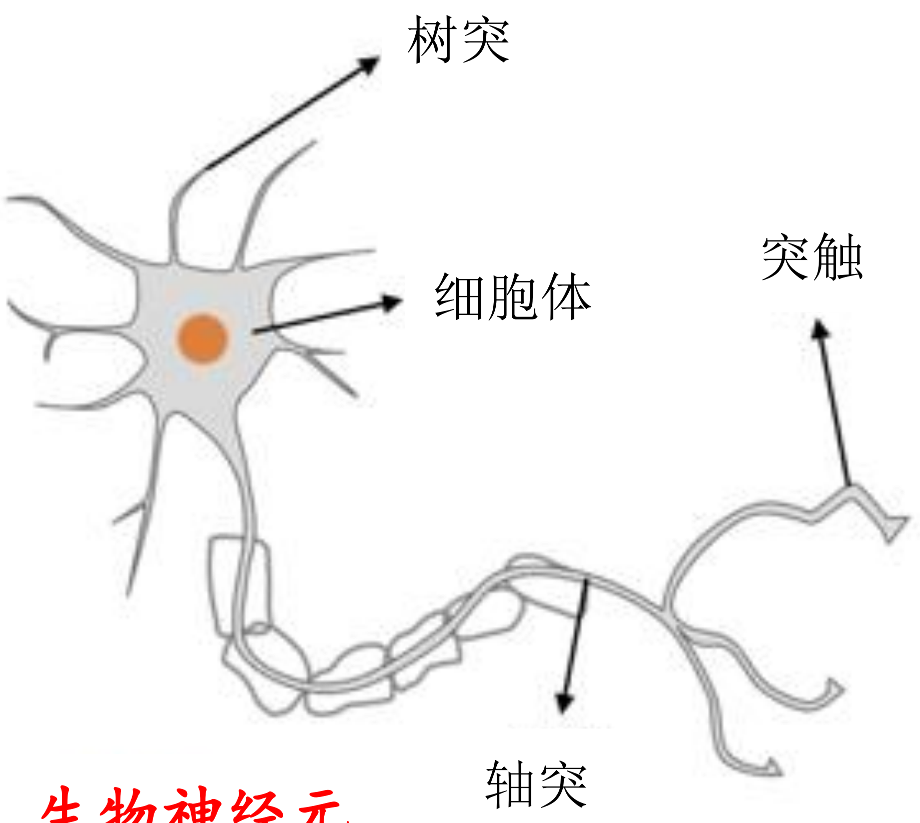
神经网络(Neural Network)

- 对输入特征向量进行多次非线性变换，再分类
- 同时完成两个工作：
  - ➢ 特征提取
  - ➢ 分类/回归

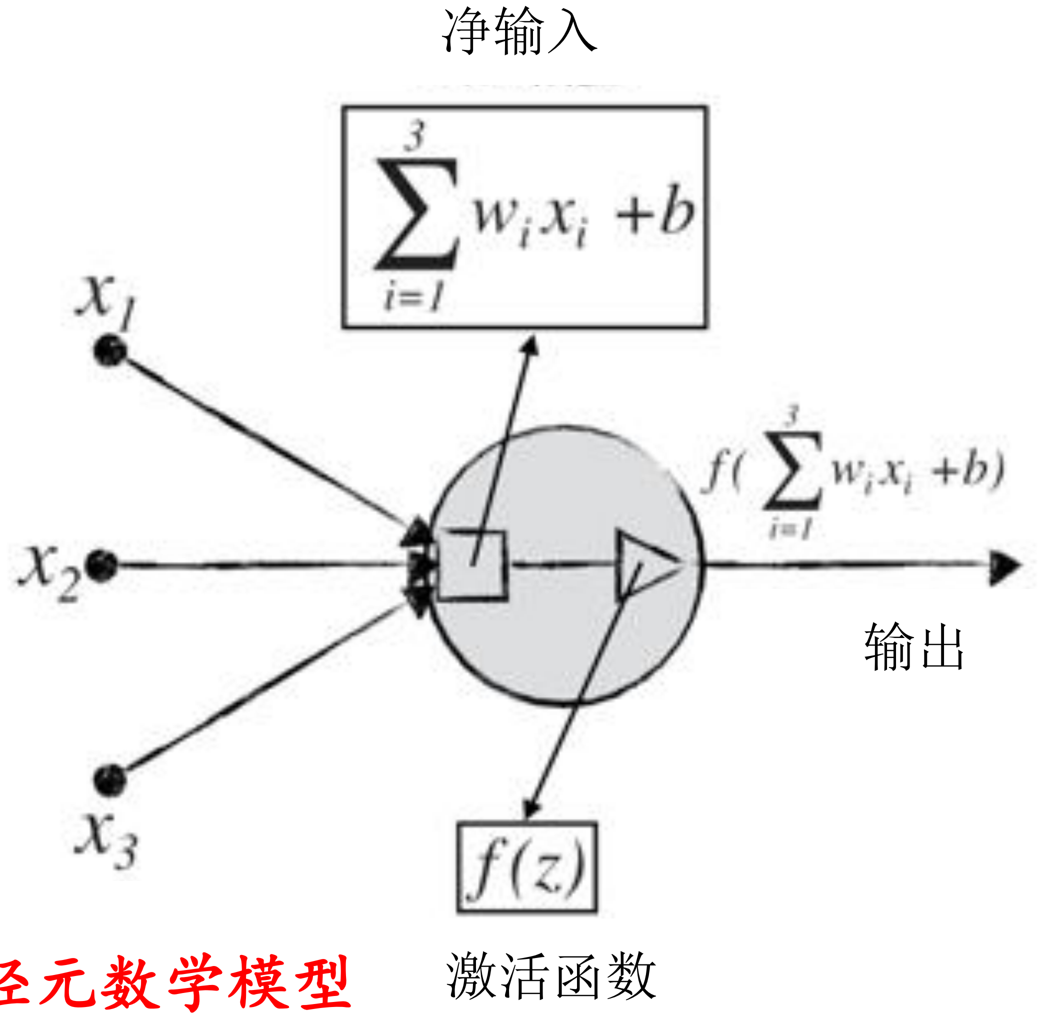


神经网络(Neural Network)

神经元



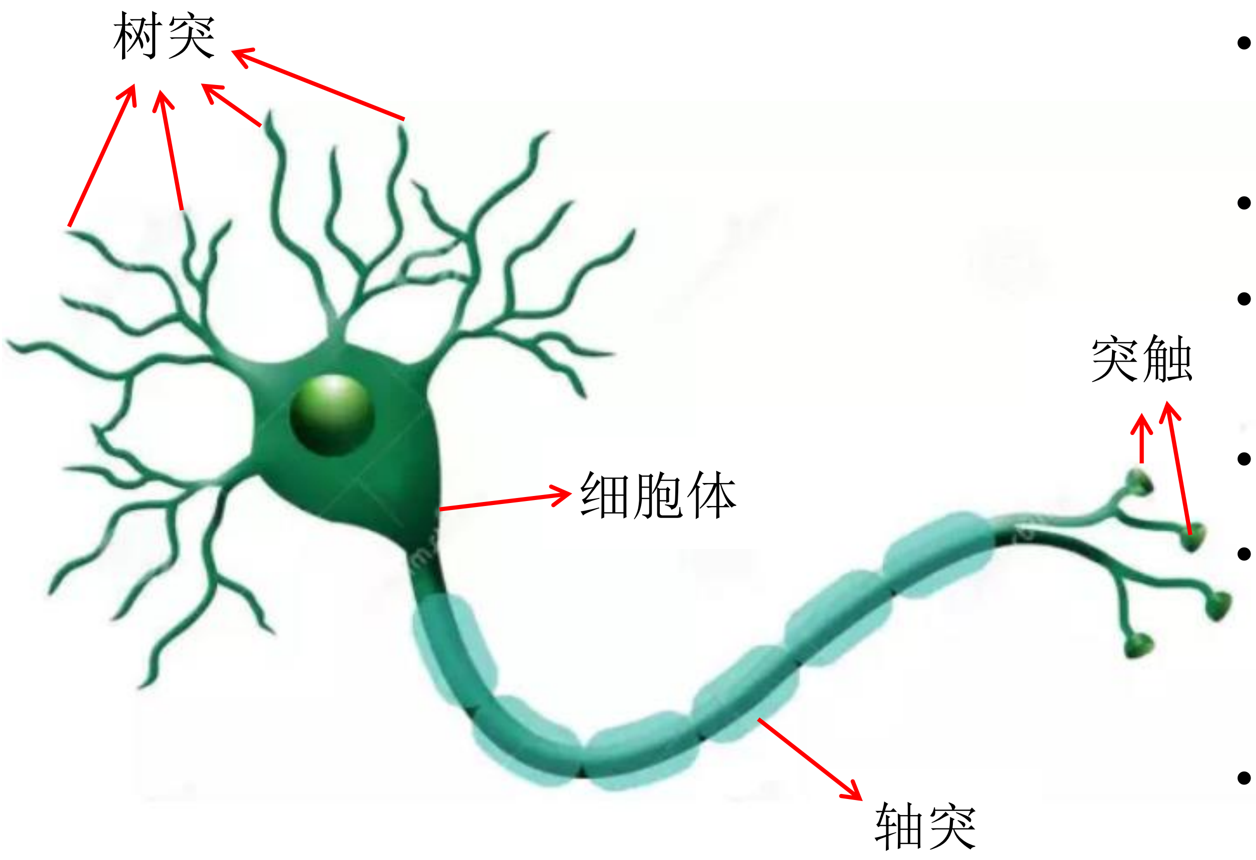
生物神经元



神经元数学模型

神经网络(Neural Network)

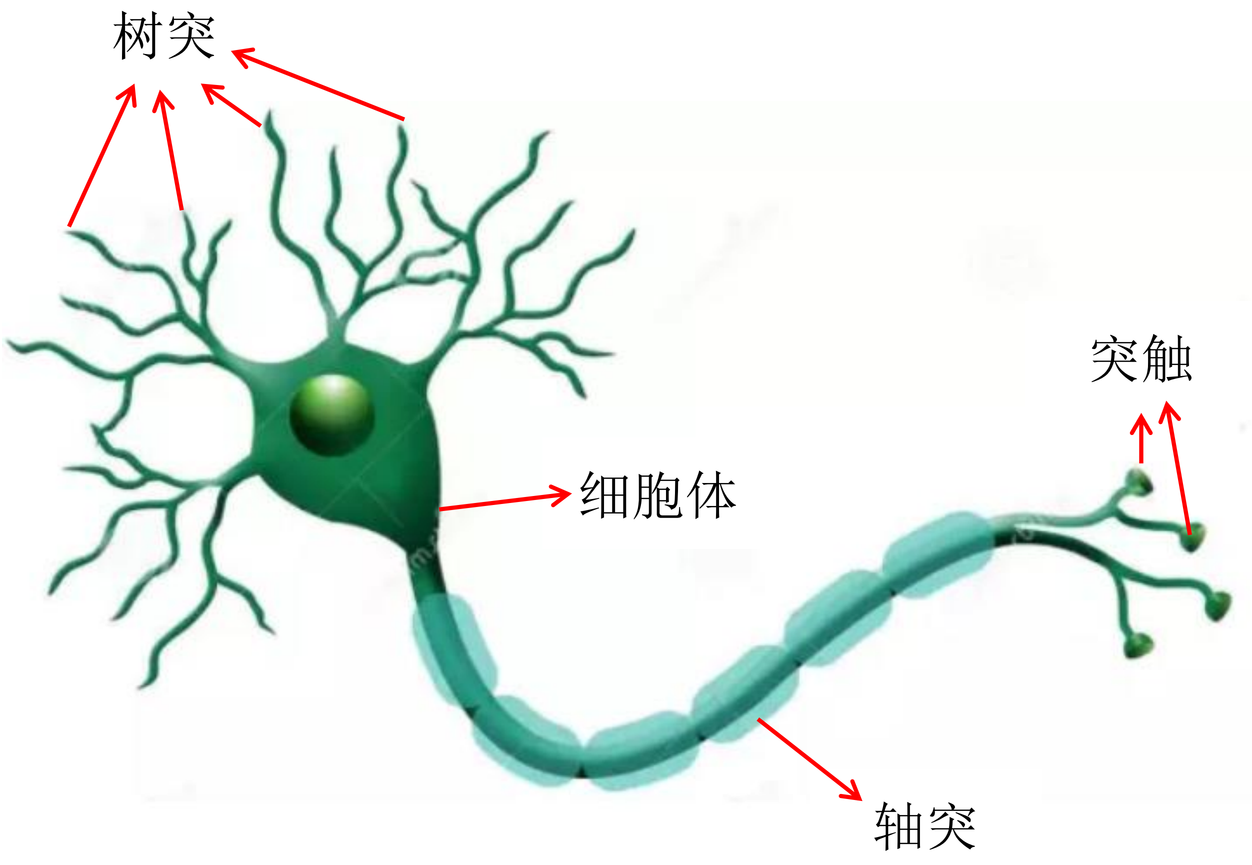
神经元



- 一个神经元细胞由树突、细胞体、轴突和突触构成
- 一个神经元有多个树突，每个树突接收来自其他神经元传递的信号
- 多个树突接收到的信号在细胞体内进行积累
- 当积累的信号超过一个门限时，轴突产生一个信号，轴突产生的信号为二元信号
- 轴突产生的信号通过突触传递给其他神经元
- 首次提出：W.S.McCulloch, W. Pitts, A logical calculus of the ideas immanent in nervous activity, Bulletin of Mathematical Biophysics, 5(4):115-133, 1943.
- 此神经元模型也被称为MCP神经元模型

The picture is from https://www.icode9.com/content-4-652630.html.

神经网络(Neural Network)



The picture is from https://www.icode9.com/content-4-652630.html.

- 树突接收到的信号：$x_1, x_2, \cdots, x_m$
- 细胞体内积累的信号 $\sum_{i=1}^{m} x_i$
- 积累信号与门限值比较。令门限值为θ。如果$\sum_{i=1}^{m} x_i > \theta$，轴突输出1；否则，轴突输出0。

$$\text{轴突输出}=\begin{cases} 1 & \text{如果} \sum_{i=1}^{m} x_i > \theta \\ 0 & \text{如果} \sum_{i=1}^{m} x_i \le \theta \end{cases}$$

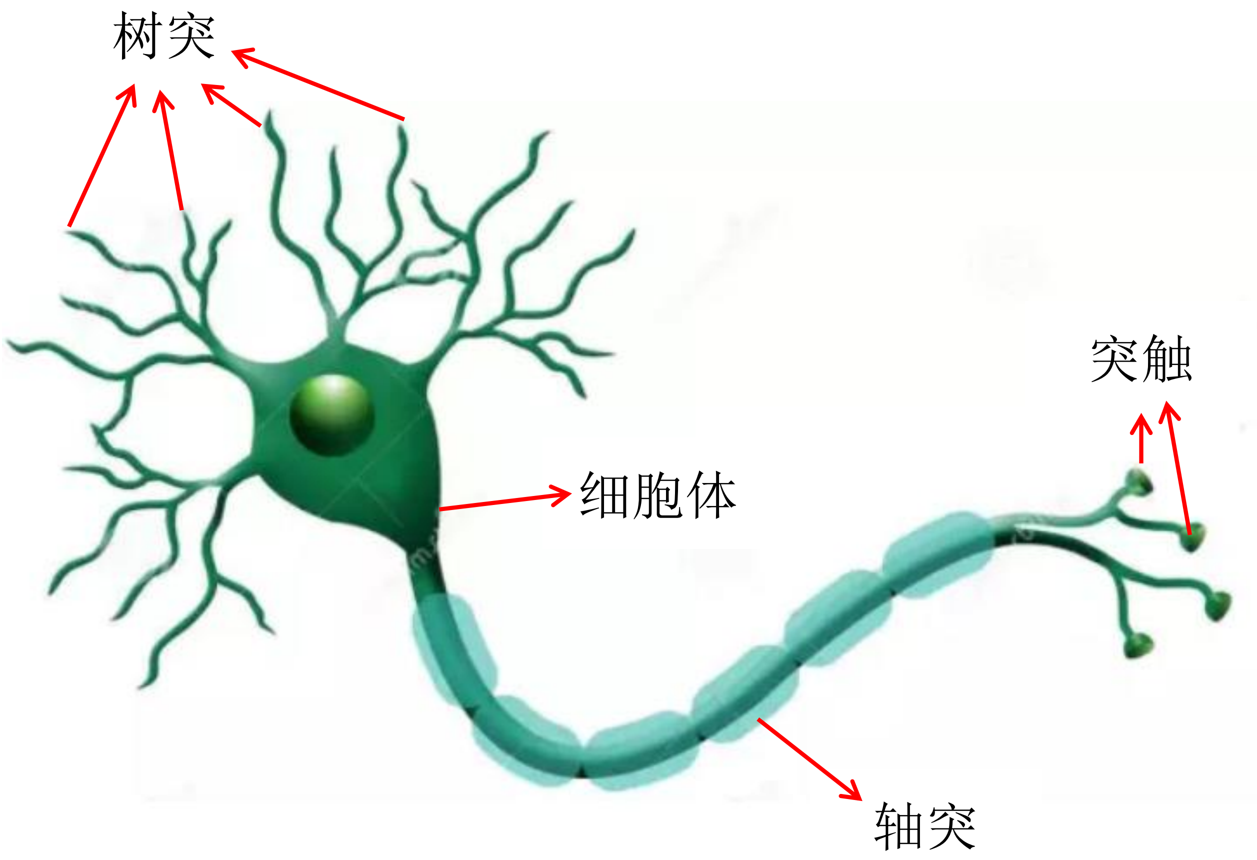
$$\text{轴突输出}=\begin{cases} 1 & \text{如果} \sum_{i=1}^{m} x_i - \theta > 0 \\ 0 & \text{如果} \sum_{i=1}^{m} x_i - \theta \le 0 \end{cases}$$

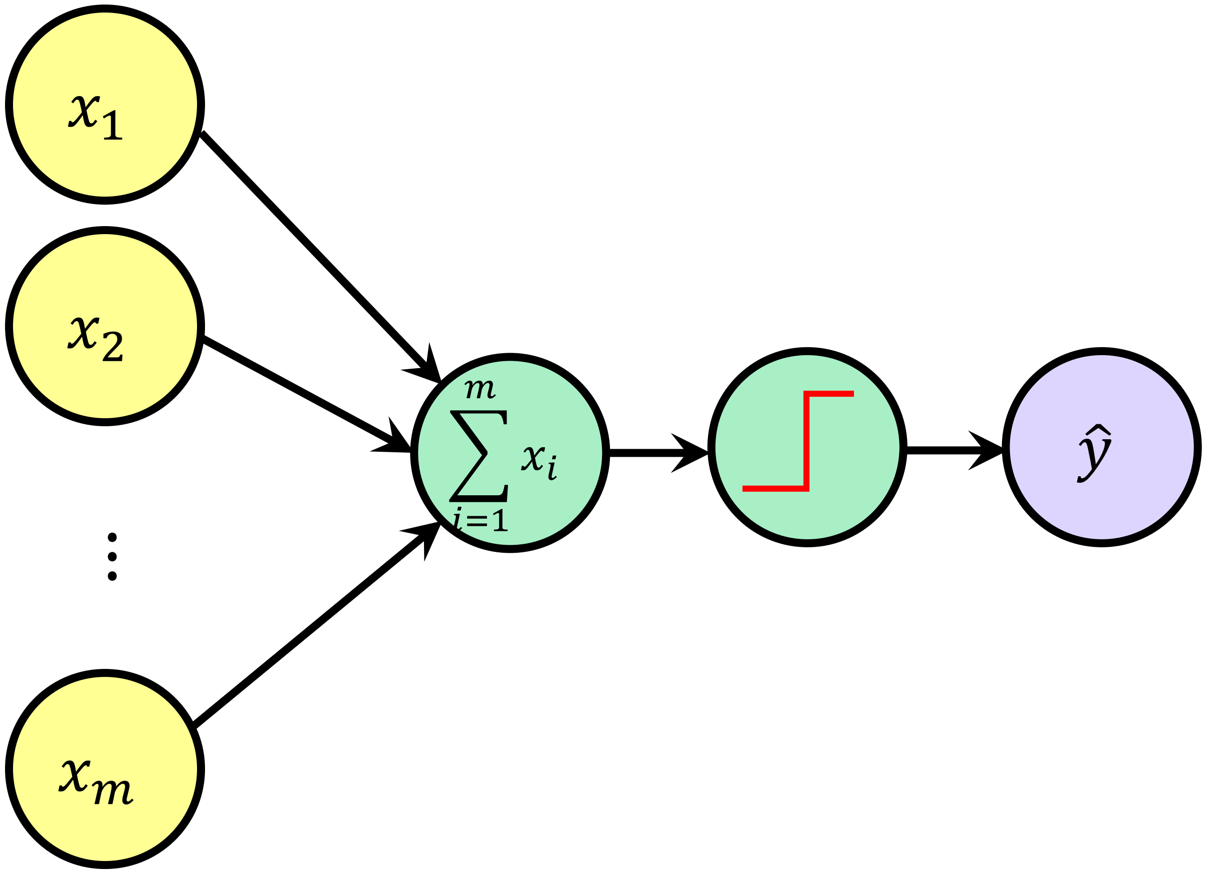
$b = -\theta$

$$\text{轴突输出}=\begin{cases} 1 & \text{如果} \sum_{i=1}^{m} x_i + b > 0 \\ 0 & \text{如果} \sum_{i=1}^{m} x_i + b \le 0 \end{cases}$$

神经网络(Neural Network)

神经元





The picture is from https://www.icode9.com/content-4-652630.html.

神经网络(Neural Network)

神经元

$$\hat{y} = \sigma(\boldsymbol{\omega}^T \boldsymbol{x} + b)$$

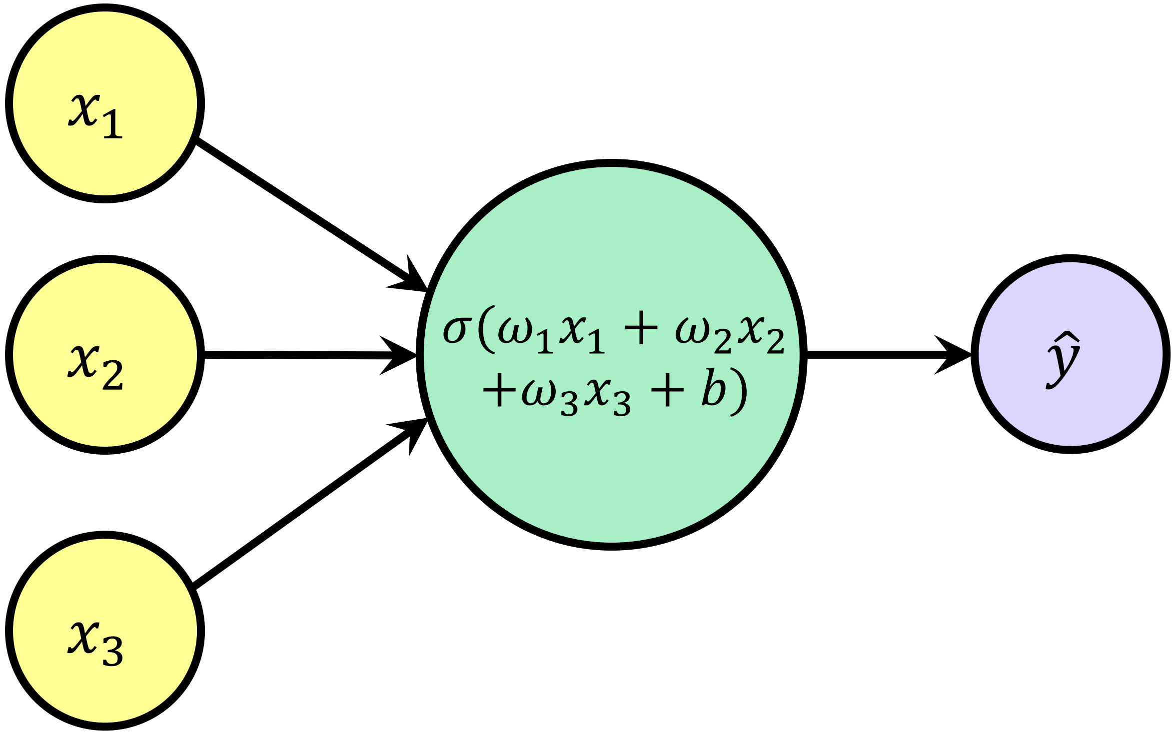
$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

- 线性变换加偏置项

$$z = \boldsymbol{\omega}^T \boldsymbol{x} + b$$

- 非线性变换

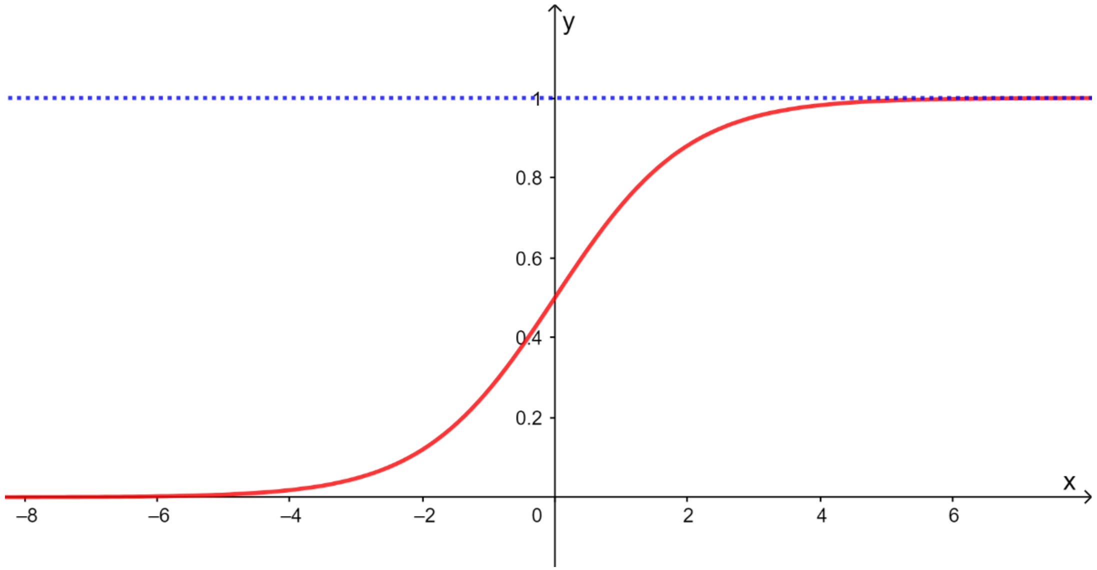
$$\hat{y} = \sigma(z)$$



神经网络(Neural Network)

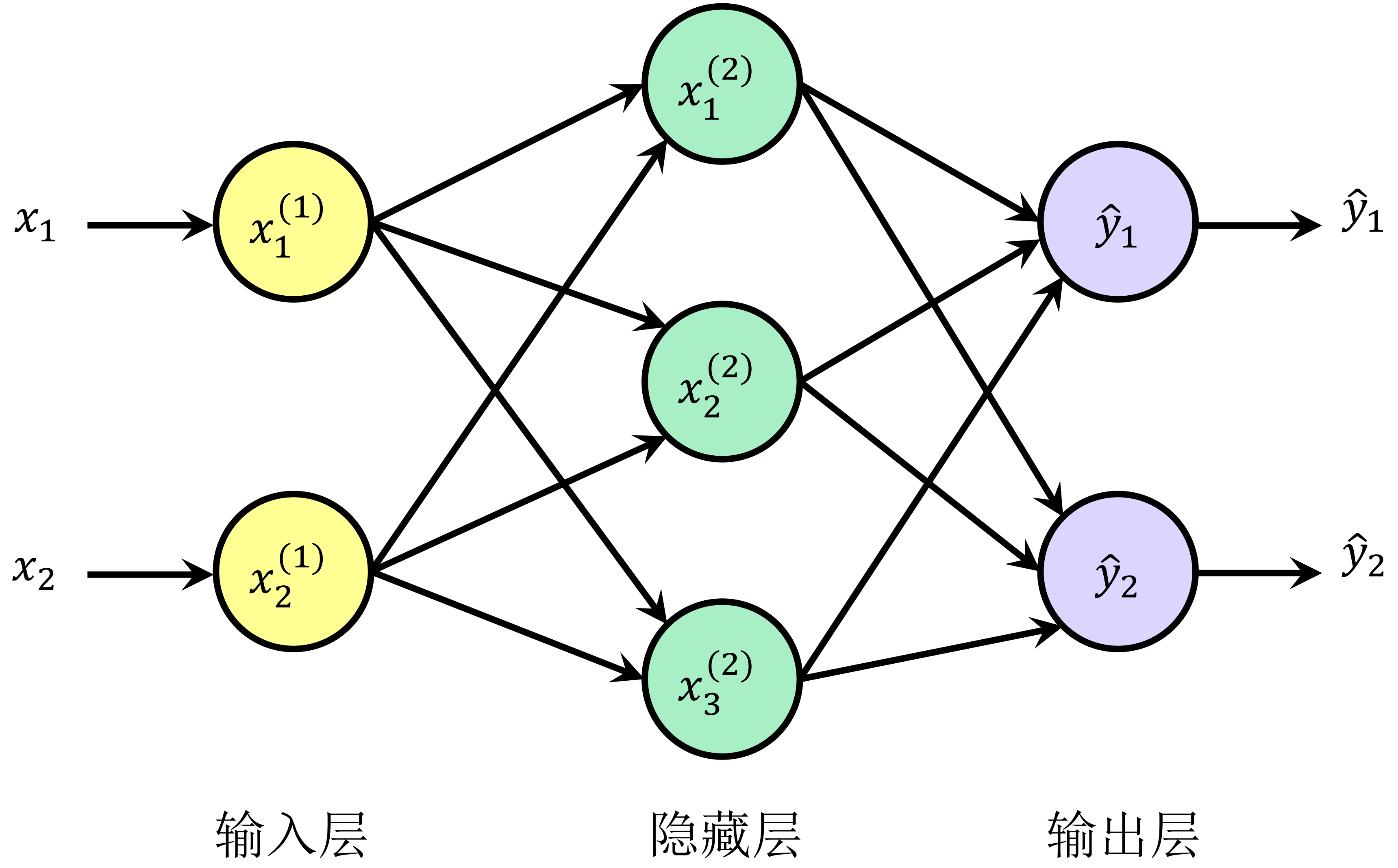
- 激活函数：逻辑函数

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



- $0 < \sigma(x) < 1$
- $\sigma(-\infty) = 0$, $\sigma(+\infty) = 1$, $\sigma(0) = 0.5$
- 逻辑函数适合表示概率

神经网络(Neural Network)



神经网络(Neural Network)

前向传播：输入层

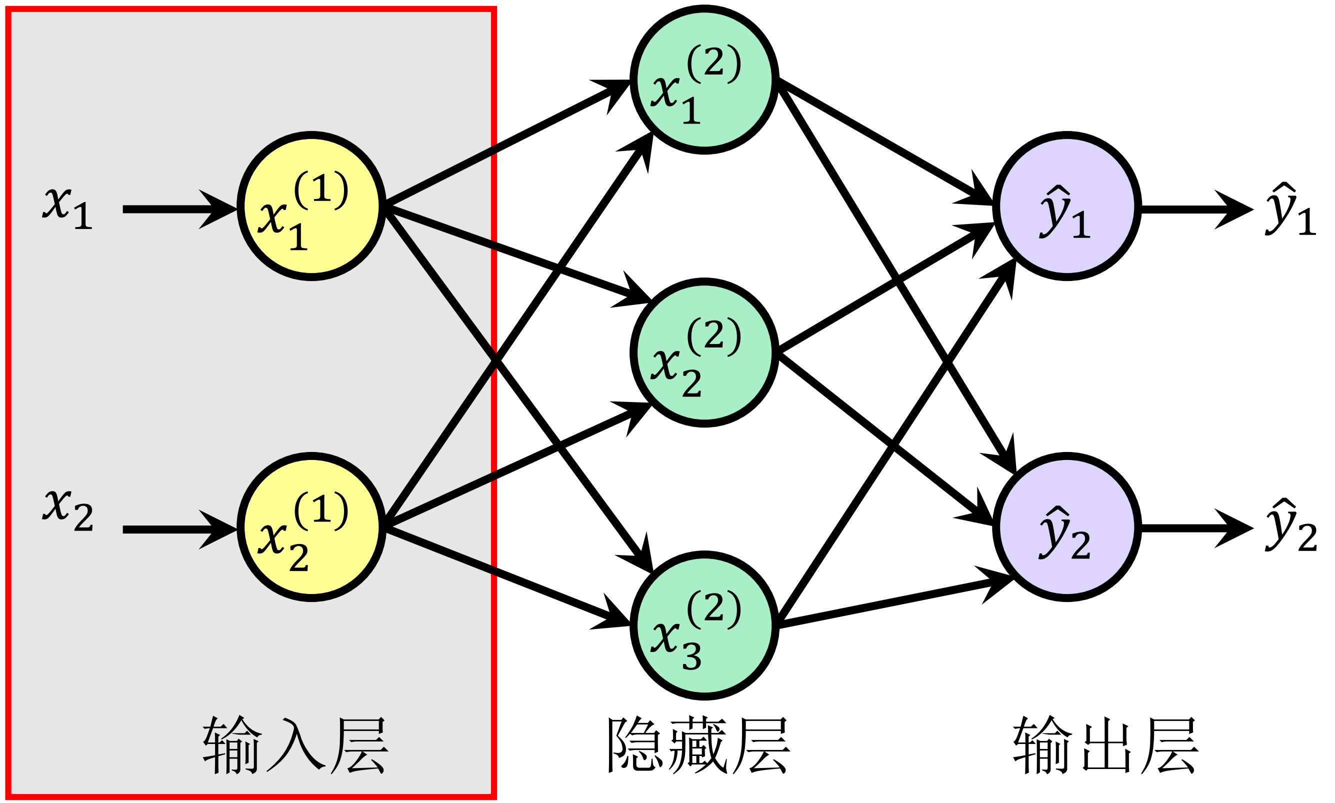
输入:

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ \\ x_2 \end{bmatrix}$$

输出:

$$x_1^{(1)} = x_1, \quad x_2^{(1)} = x_2$$

$$\boldsymbol{x}^{(1)} = \begin{bmatrix} x_1^{(1)} \\ \\ x_2^{(1)} \end{bmatrix}$$



输入层完成一个拷贝工作.

神经网络(Neural Network)

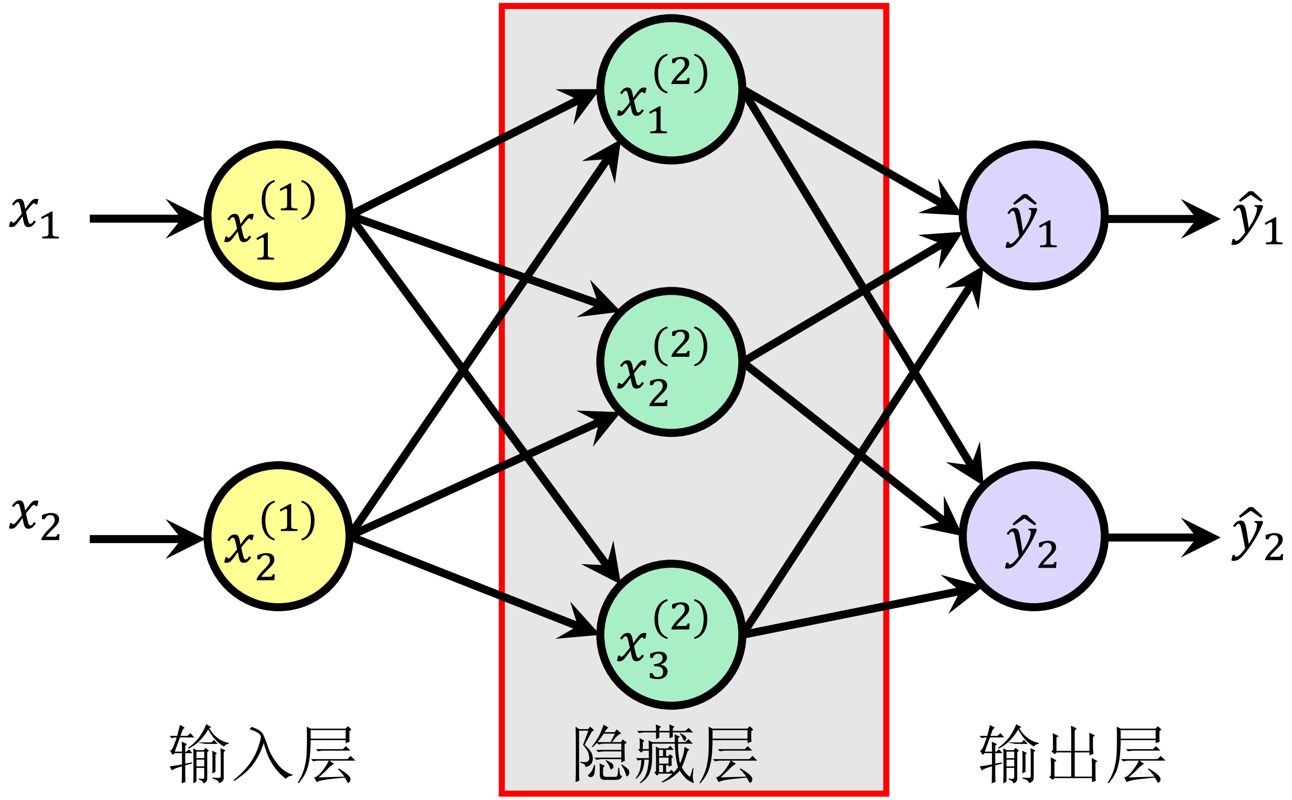
前向传播：隐藏层

输入:

$$\begin{cases} z_1^{(2)} = W_{11}^{(1)} x_1^{(1)} + W_{12}^{(1)} x_2^{(1)} + b_1^{(1)} \\ z_2^{(2)} = W_{21}^{(1)} x_1^{(1)} + W_{22}^{(1)} x_2^{(1)} + b_2^{(1)} \\ z_3^{(2)} = W_{31}^{(1)} x_1^{(1)} + W_{32}^{(1)} x_2^{(1)} + b_3^{(1)} \end{cases}$$

$$\begin{bmatrix} z_1^{(2)} \\ z_2^{(2)} \\ z_3^{(2)} \end{bmatrix} = \begin{bmatrix} W_{11}^{(1)} & W_{12}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} \\ W_{31}^{(1)} & W_{32}^{(1)} \end{bmatrix} \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \end{bmatrix} + \begin{bmatrix} b_1^{(1)} \\ b_2^{(1)} \\ b_3^{(1)} \end{bmatrix}$$

$$\boldsymbol{z}^{(2)} = \boldsymbol{W}^{(1)} \boldsymbol{x}^{(1)} + \boldsymbol{b}^{(1)}$$



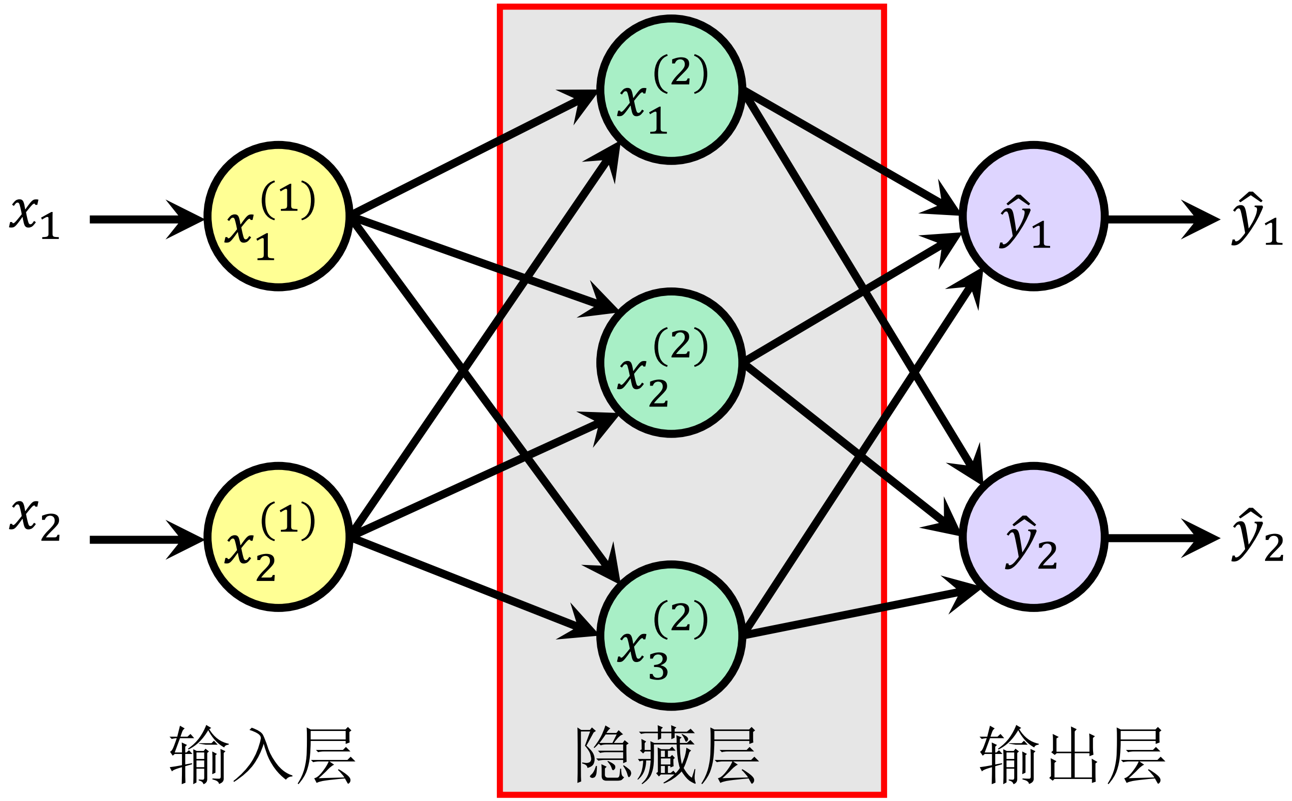
神经网络(Neural Network)

前向传播：隐藏层

输出:

$$\begin{cases} x_1^{(2)} = \sigma(z_1^{(2)}) \\ x_2^{(2)} = \sigma(z_2^{(2)}) \\ x_3^{(2)} = \sigma(z_3^{(2)}) \end{cases}$$

$$\boldsymbol{x}^{(2)} = \sigma(\boldsymbol{z}^{(2)})$$



神经网络(Neural Network)

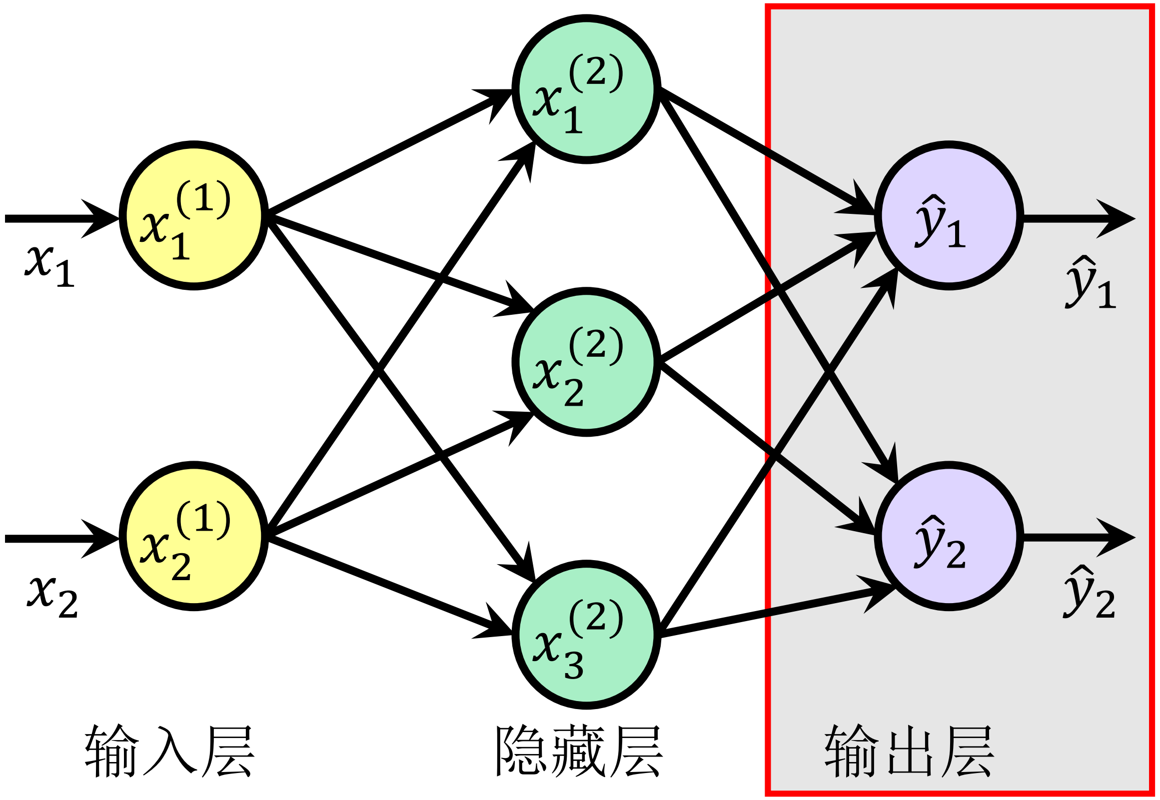
前向传播：输出层

输入:

$$\begin{cases} z_1^{(3)} = W_{11}^{(2)} x_1^{(2)} + W_{12}^{(2)} x_2^{(2)} + W_{13}^{(2)} x_3^{(2)} + b_1^{(2)} \\ z_2^{(3)} = W_{21}^{(2)} x_1^{(2)} + W_{22}^{(2)} x_2^{(2)} + W_{13}^{(2)} x_3^{(2)} + b_2^{(2)} \end{cases}$$

$$\begin{bmatrix} z_1^{(3)} \\ z_2^{(3)} \end{bmatrix} = \begin{bmatrix} W_{11}^{(2)} & W_{12}^{(2)} & W_{13}^{(2)} \\ W_{21}^{(2)} & W_{22}^{(2)} & W_{23}^{(2)} \end{bmatrix} \begin{bmatrix} x_1^{(2)} \\ x_2^{(2)} \\ x_3^{(2)} \end{bmatrix} + \begin{bmatrix} b_1^{(2)} \\ b_2^{(2)} \end{bmatrix}$$

$$\boldsymbol{z}^{(3)} = \boldsymbol{W}^{(2)} \boldsymbol{x}^{(2)} + \boldsymbol{b}^{(2)}$$



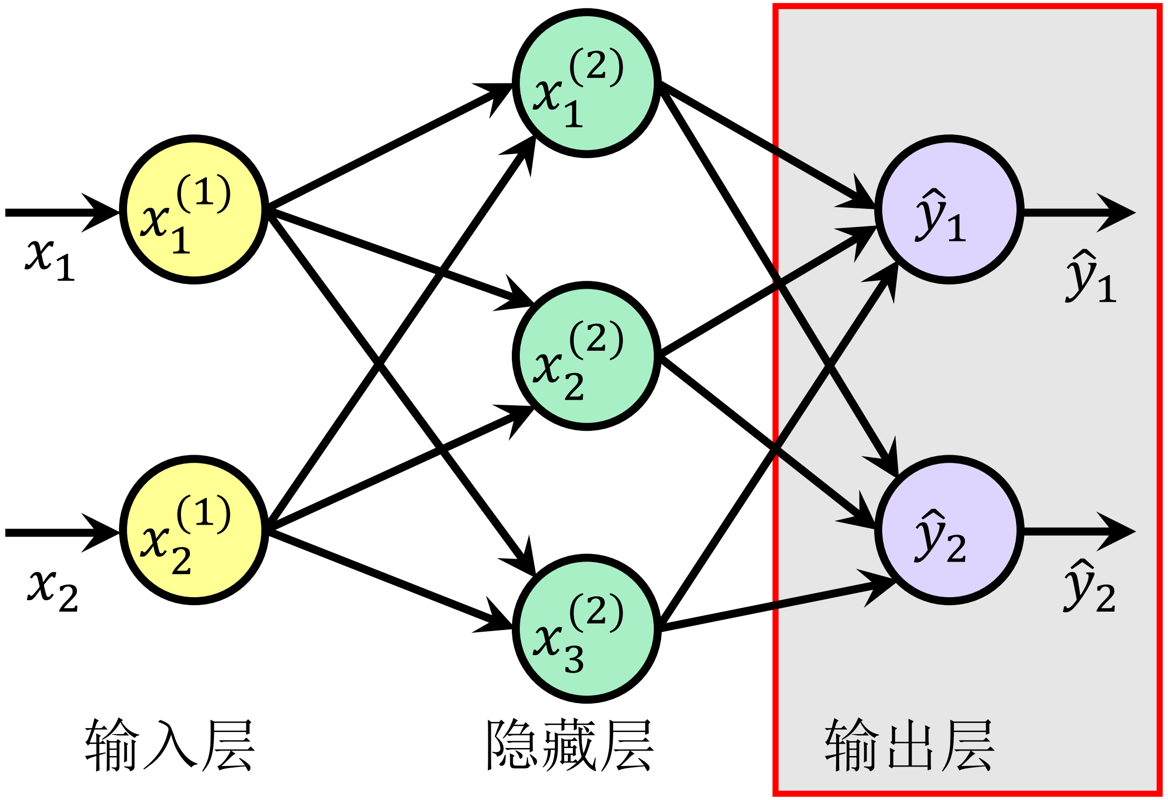
神经网络(Neural Network)

前向传播：输出层

输出:

$$
\begin{cases}
\hat{y}_1 = \dfrac{e^{z_1^{(3)}}}{e^{z_1^{(3)}} + e^{z_2^{(3)}}} \\
\hat{y}_2 = \dfrac{e^{z_2^{(3)}}}{e^{z_1^{(3)}} + e^{z_2^{(3)}}}
\end{cases}
$$

$$
\hat{\boldsymbol{y}} = \begin{bmatrix} \hat{y}_1 \\ \\ \hat{y}_2 \end{bmatrix}
$$



神经网络(Neural Network)

前向传播

- 输入层

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}$$

- 隐藏层

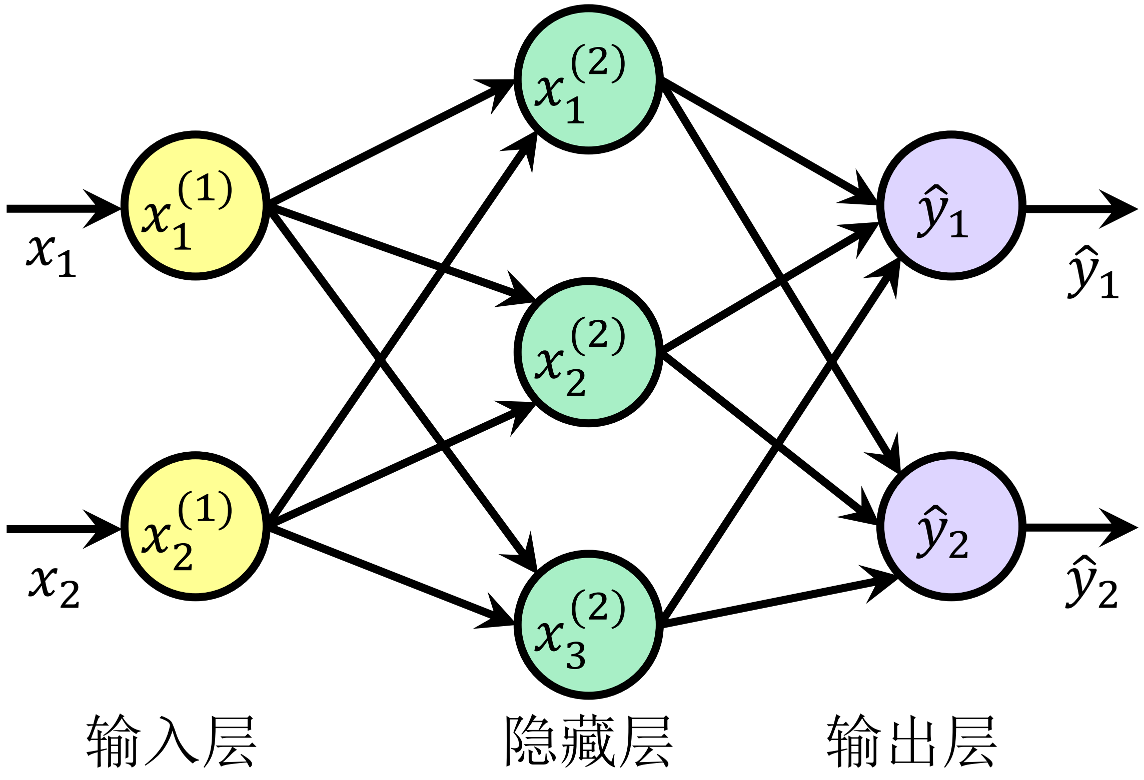
$$\boldsymbol{z}^{(2)} = \boldsymbol{W}^{(1)}\boldsymbol{x}^{(1)} + \boldsymbol{b}^{(1)}$$
$$\boldsymbol{x}^{(2)} = \sigma(\boldsymbol{z}^{(2)})$$

- 输出层

$$\boldsymbol{z}^{(3)} = \boldsymbol{W}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{b}^{(2)}$$
$$\hat{\boldsymbol{y}} = \boldsymbol{\sigma}(\boldsymbol{z}^{(3)})$$

- 损失函数(交叉熵损失函数)

$$L = -y_1\log(\hat{y}_1) - y_2\log(\hat{y}_2)$$



神经网络(Neural Network)

反向传播

- 损失函数

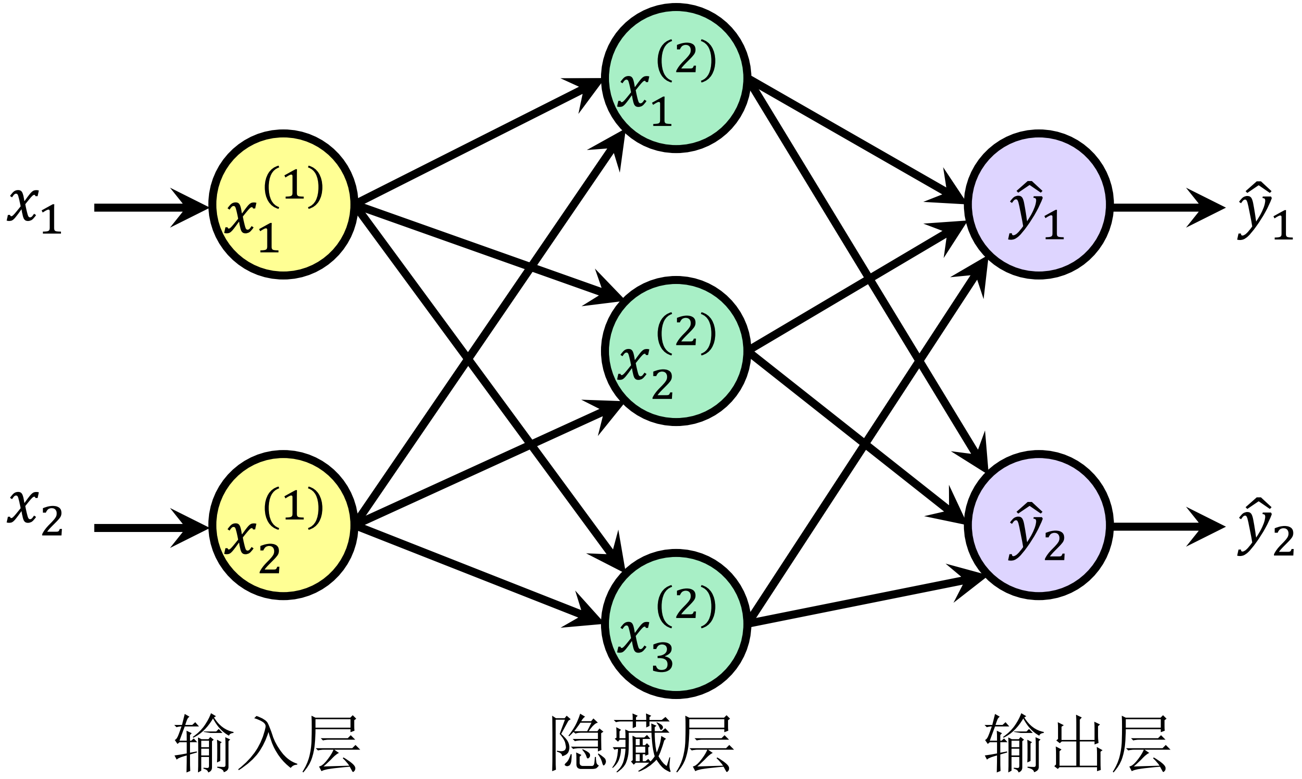
$$L = -y_1 \log(\hat{y}_1) - y_2 \log(\hat{y}_2)$$

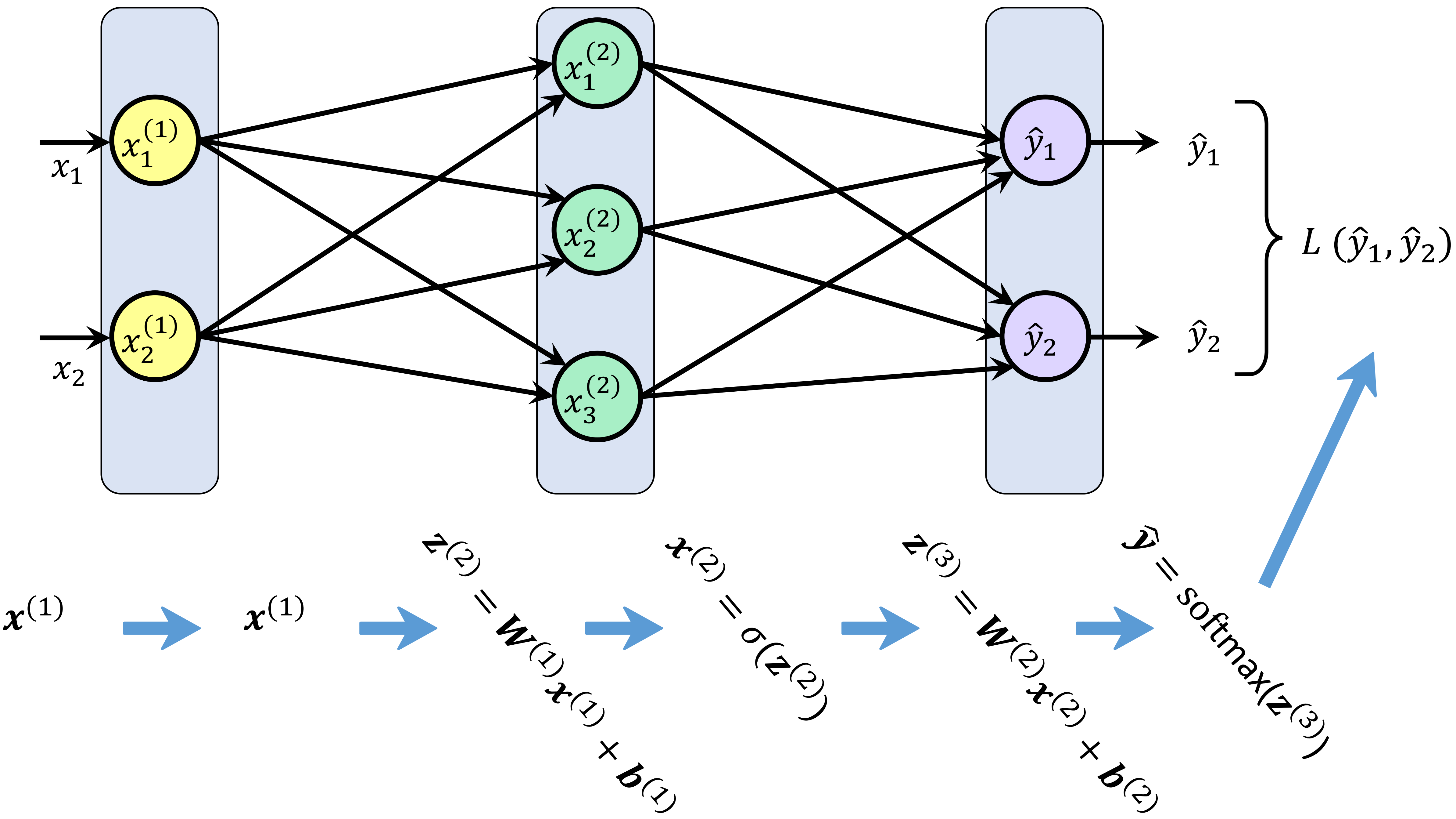
- 参数

$$\boldsymbol{\theta} = \{\boldsymbol{W}^{(1)}, \boldsymbol{b}^{(1)}, \boldsymbol{W}^{(2)}, \boldsymbol{b}^{(1)}\}$$

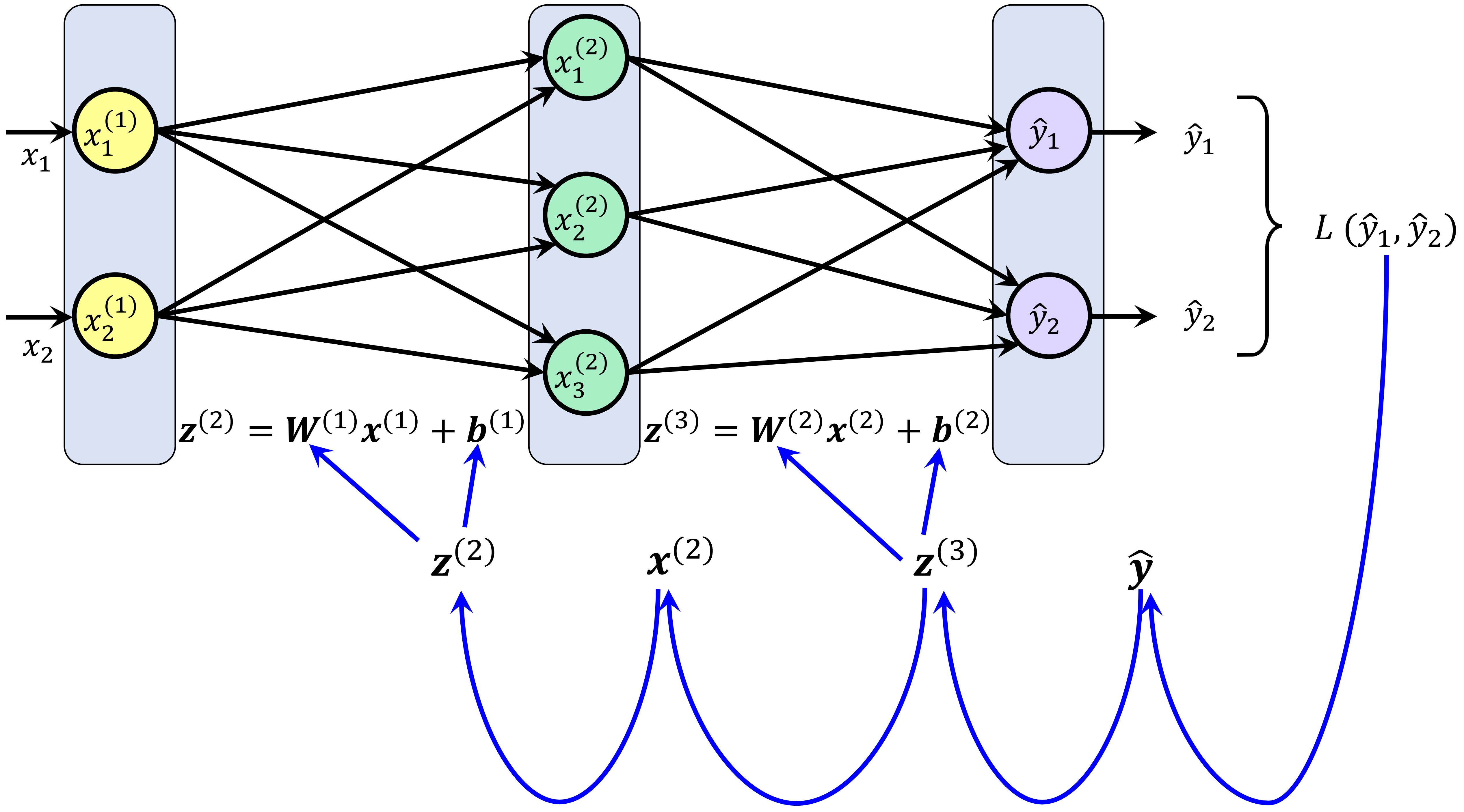
- 梯度下降法最小化损失函数

$$\boldsymbol{\theta}_i \leftarrow \boldsymbol{\theta}_{i-1} - \eta \left.\frac{\partial L}{\partial \boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{i-1}}$$





x_1
x_2
$x_1^{(1)}$
$x_2^{(1)}$
$x_1^{(2)}$
$x_2^{(2)}$
$x_3^{(2)}$
$\hat{y}_1$
$\hat{y}_2$
$\hat{y}_1$
$\hat{y}_2$
$L\left(\hat{y}_1, \hat{y}_2\right)$
$\boldsymbol{x}^{(1)}$
$\boldsymbol{x}^{(1)}$
$\boldsymbol{z}^{(2)} = \boldsymbol{W}^{(1)}\boldsymbol{x}^{(1)} + \boldsymbol{b}^{(1)}$
$\boldsymbol{x}^{(2)} = \sigma(\boldsymbol{z}^{(2)})$
$\boldsymbol{z}^{(3)} = \boldsymbol{W}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{b}^{(2)}$
$\hat{\boldsymbol{y}} = \text{softmax}(\boldsymbol{z}^{(3)})$

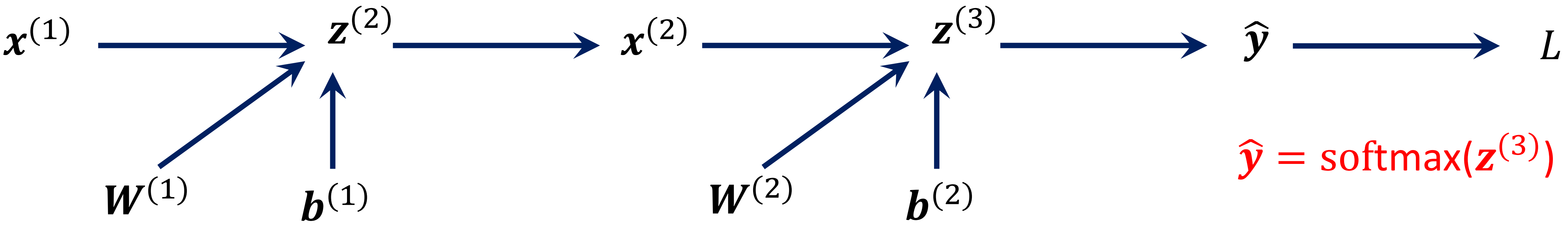


x_1
x_2
$x_1^{(1)}$
$x_2^{(1)}$
$x_1^{(2)}$
$x_2^{(2)}$
$x_3^{(2)}$
$\hat{y}_1$
$\hat{y}_2$
$\hat{y}_1$
$\hat{y}_2$
$L\left(\hat{y}_1, \hat{y}_2\right)$
$\boldsymbol{z}^{(2)} = \boldsymbol{W}^{(1)}\boldsymbol{x}^{(1)} + \boldsymbol{b}^{(1)}$
$\boldsymbol{z}^{(3)} = \boldsymbol{W}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{b}^{(2)}$
$\boldsymbol{z}^{(2)}$
$\boldsymbol{x}^{(2)}$
$\boldsymbol{z}^{(3)}$
$\hat{\boldsymbol{y}}$

神经网络(Neural Network)

反向传播

$$\boldsymbol{x}^{(2)} = \sigma(\boldsymbol{z}^{(2)}) \qquad \boldsymbol{z}^{(3)} = \boldsymbol{W}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{b}^{(2)}$$



$$\hat{\boldsymbol{y}} = \mathrm{softmax}(\boldsymbol{z}^{(3)})$$

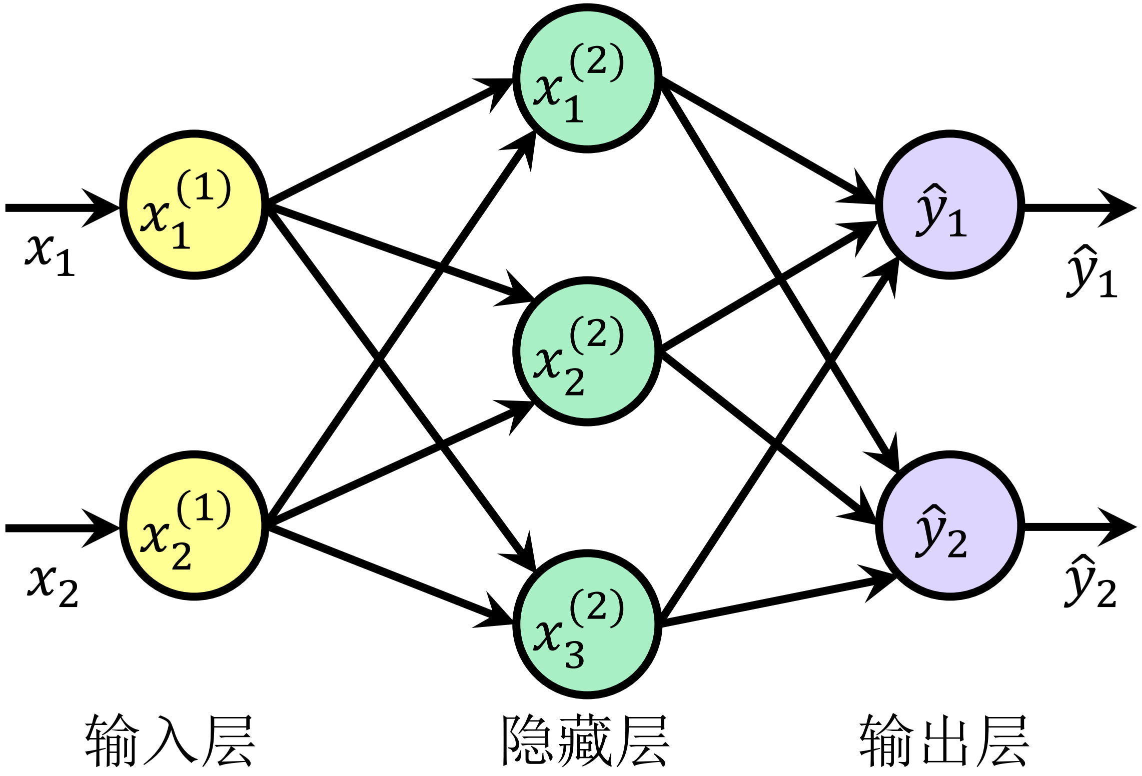
$$\boldsymbol{z}^{(2)} = \boldsymbol{W}^{(1)}\boldsymbol{x}^{(1)} + \boldsymbol{b}^{(1)}$$

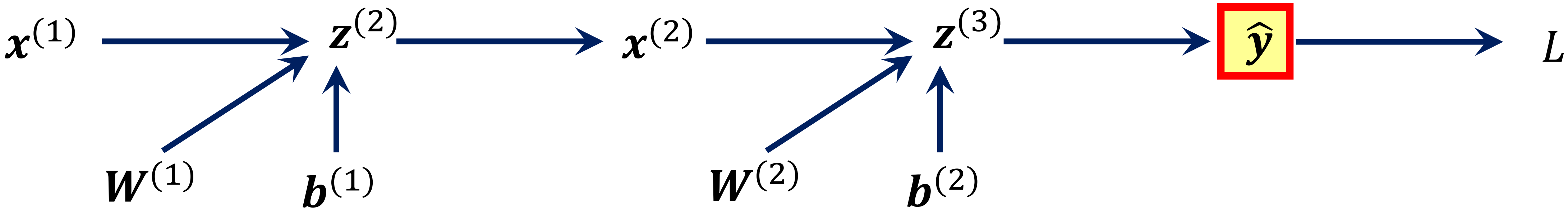
神经网络(Neural Network)

反向传播

$$L = -y_1 \log(\hat{y}_1) - y_2 \log(\hat{y}_2)$$

$$\frac{\partial L}{\partial \hat{\boldsymbol{y}}} = \begin{bmatrix} \frac{\partial L}{\partial \hat{y}_1} \\ \frac{\partial L}{\partial \hat{y}_2} \end{bmatrix} = \begin{bmatrix} \frac{-y_1}{\hat{y}_1} \\ \frac{-y_2}{\hat{y}_2} \end{bmatrix}$$



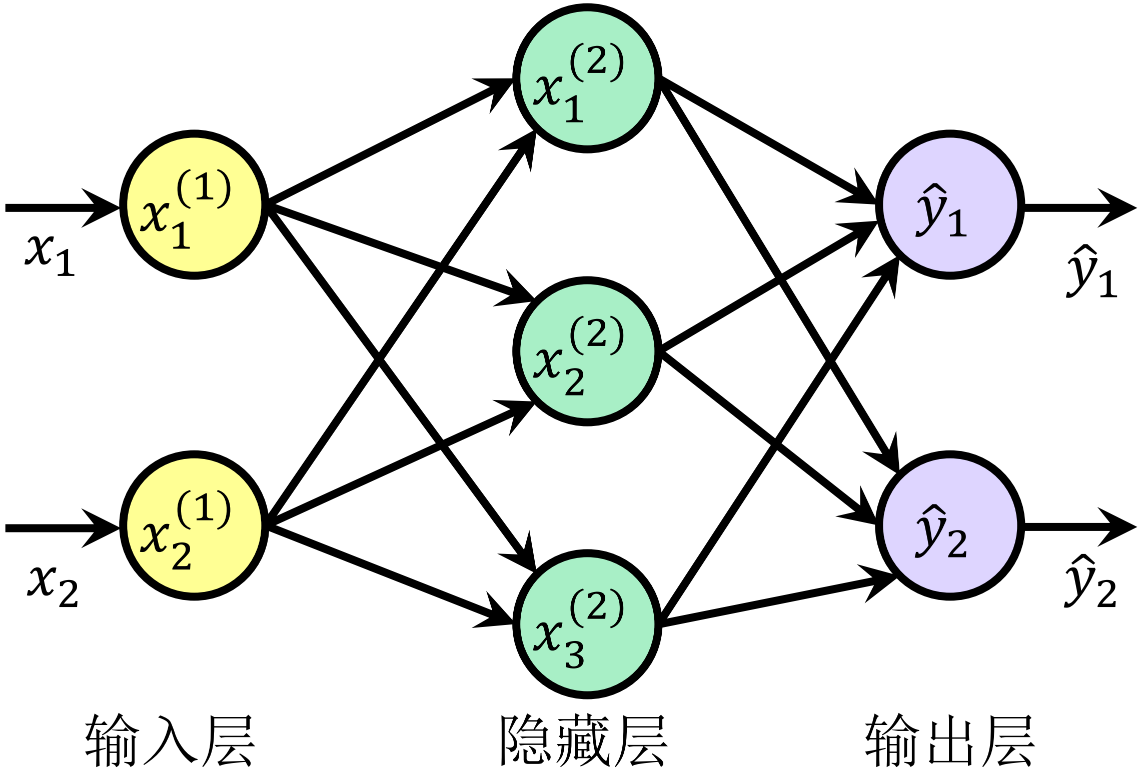


神经网络(Neural Network)

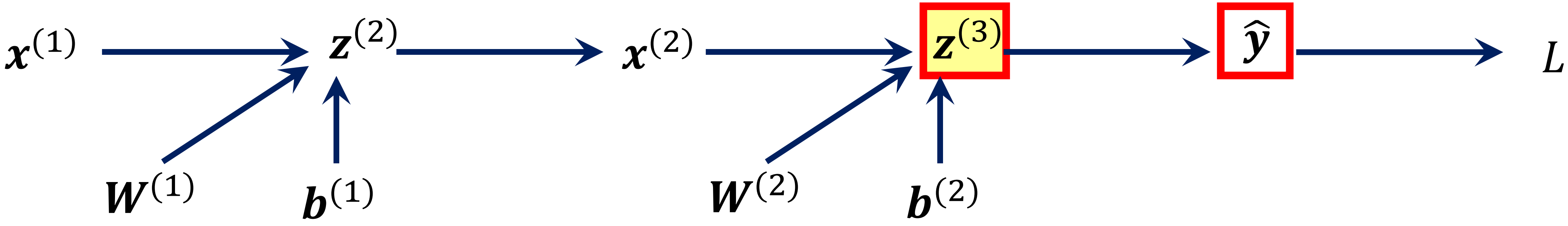
反向传播

$$\frac{\partial L}{\partial \boldsymbol{z}^{(3)}} = \frac{\partial \widehat{\boldsymbol{y}}}{\partial \boldsymbol{z}^{(3)}} \frac{\partial L}{\partial \widehat{\boldsymbol{y}}}$$

$$\begin{cases} \hat{y}_1 = \dfrac{e^{z_1^{(3)}}}{e^{z_1^{(3)}} + e^{z_2^{(3)}}} \\ \hat{y}_2 = \dfrac{e^{z_2^{(3)}}}{e^{z_1^{(3)}} + e^{z_2^{(3)}}} \end{cases}$$



$$\frac{\partial \widehat{\boldsymbol{y}}}{\partial \boldsymbol{z}^{(3)}} = \begin{bmatrix} \hat{y}_1\hat{y}_2 & -\hat{y}_1\hat{y}_2 \\ -\hat{y}_1\hat{y}_2 & \hat{y}_1\hat{y}_2 \end{bmatrix}$$

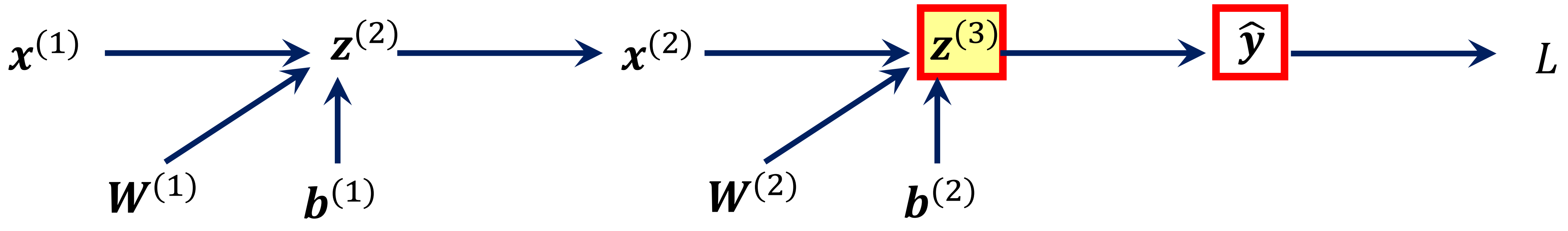


神经网络(Neural Network)

反向传播

$$\frac{\partial L}{\partial \boldsymbol{z}^{(3)}} = \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}^{(3)}} \frac{\partial L}{\partial \hat{\boldsymbol{y}}}, \qquad \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}^{(3)}} = \begin{bmatrix} \hat{y}_1 \hat{y}_2 & -\hat{y}_1 \hat{y}_2 \\ -\hat{y}_1 \hat{y}_2 & \hat{y}_1 \hat{y}_2 \end{bmatrix}, \qquad \frac{\partial L}{\partial \hat{\boldsymbol{y}}} = \begin{bmatrix} \frac{-y_1}{\hat{y}_1} \\ \frac{-y_2}{\hat{y}_2} \end{bmatrix}$$

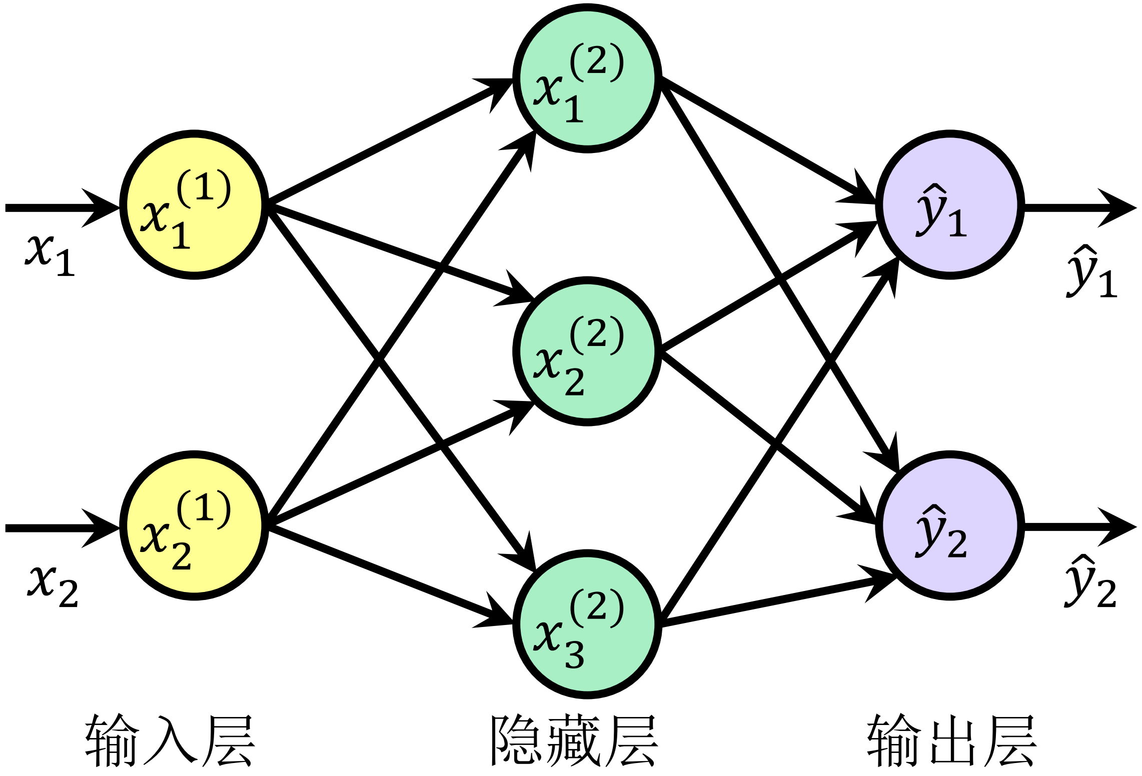
$$\frac{\partial L}{\partial \boldsymbol{z}^{(3)}} = \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}^{(3)}} \frac{\partial L}{\partial \hat{\boldsymbol{y}}} = \begin{bmatrix} \hat{y}_1 \hat{y}_2 & -\hat{y}_1 \hat{y}_2 \\ -\hat{y}_1 \hat{y}_2 & \hat{y}_1 \hat{y}_2 \end{bmatrix} \begin{bmatrix} \frac{-y_1}{\hat{y}_1} \\ \frac{-y_2}{\hat{y}_2} \end{bmatrix} = \begin{bmatrix} -y_1 \hat{y}_2 + \hat{y}_1 y_2 \\ y_1 \hat{y}_2 - \hat{y}_1 y_2 \end{bmatrix} = \begin{bmatrix} \hat{y}_1 - y_1 \\ \hat{y}_2 - y_2 \end{bmatrix} = \hat{\boldsymbol{y}} - \boldsymbol{y}$$

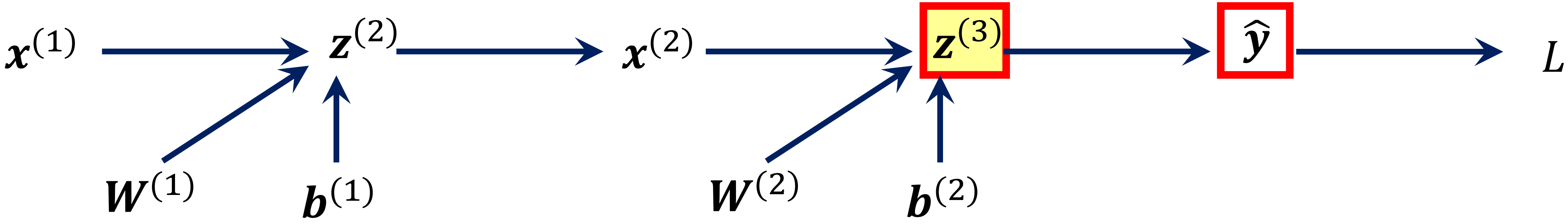


神经网络(Neural Network)

反向传播

$$\frac{\partial L}{\partial \boldsymbol{z}^{(3)}} = \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}^{(3)}} \frac{\partial L}{\partial \hat{\boldsymbol{y}}} = \hat{\boldsymbol{y}} - \boldsymbol{y}$$



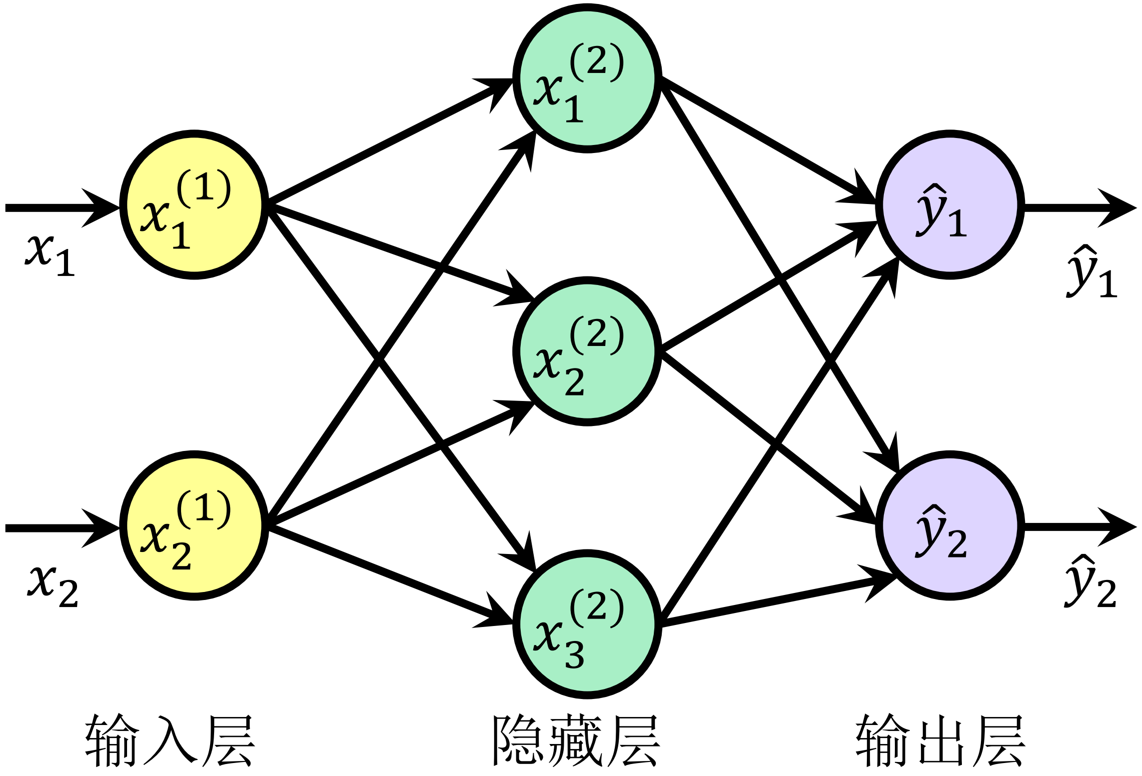


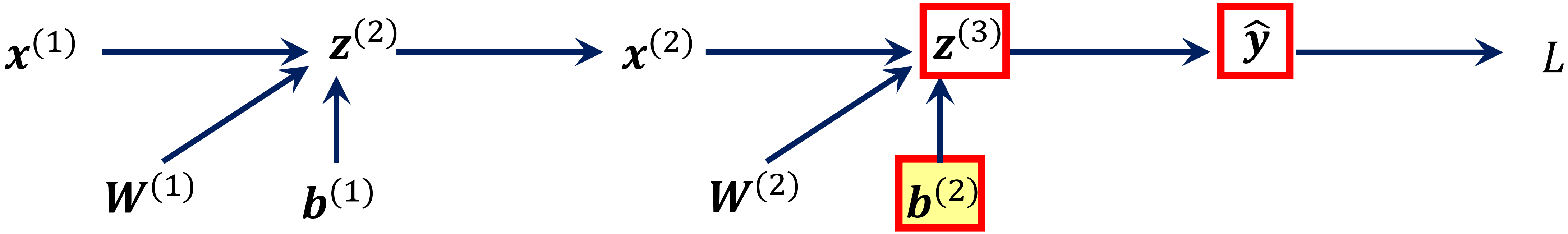
神经网络(Neural Network)

反向传播

$$\boldsymbol{z}^{(3)} = \boldsymbol{W}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{b}^{(2)}$$

$$\frac{\partial L}{\partial \boldsymbol{b}^{(2)}} = \frac{\partial \boldsymbol{z}^{(3)}}{\partial \boldsymbol{b}^{(2)}}\frac{\partial L}{\partial \boldsymbol{z}^{(3)}} = \boldsymbol{I}\frac{\partial L}{\partial \boldsymbol{z}^{(3)}} = \frac{\partial L}{\partial \boldsymbol{z}^{(3)}}$$



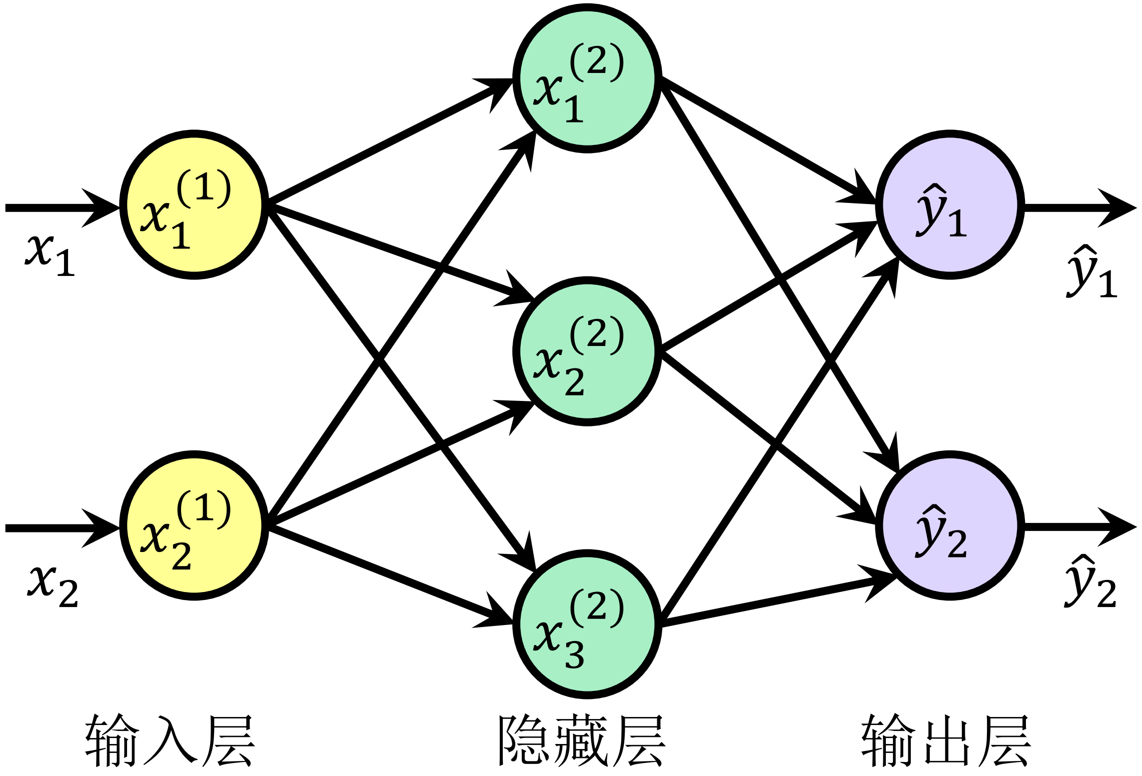


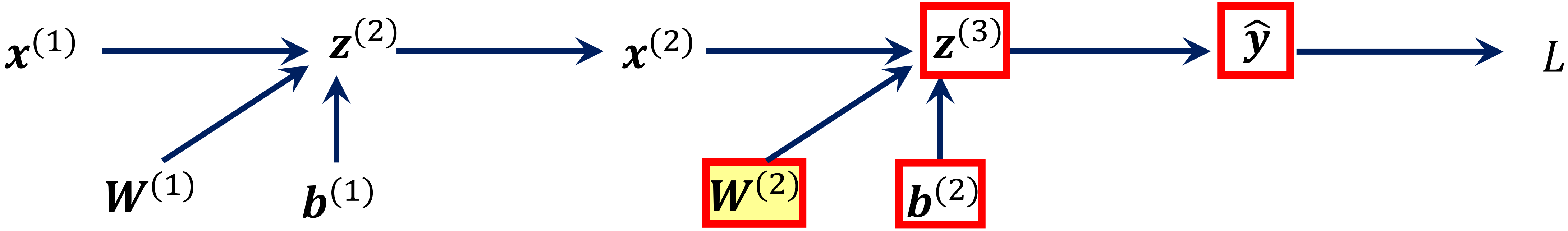
神经网络(Neural Network)

反向传播

$$\boldsymbol{z}^{(3)} = \boldsymbol{W}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{b}^{(2)}$$

$$\frac{\partial L}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial \boldsymbol{z}^{(3)}}{\partial \boldsymbol{W}^{(2)}} \frac{\partial L}{\partial \boldsymbol{z}^{(3)}}$$





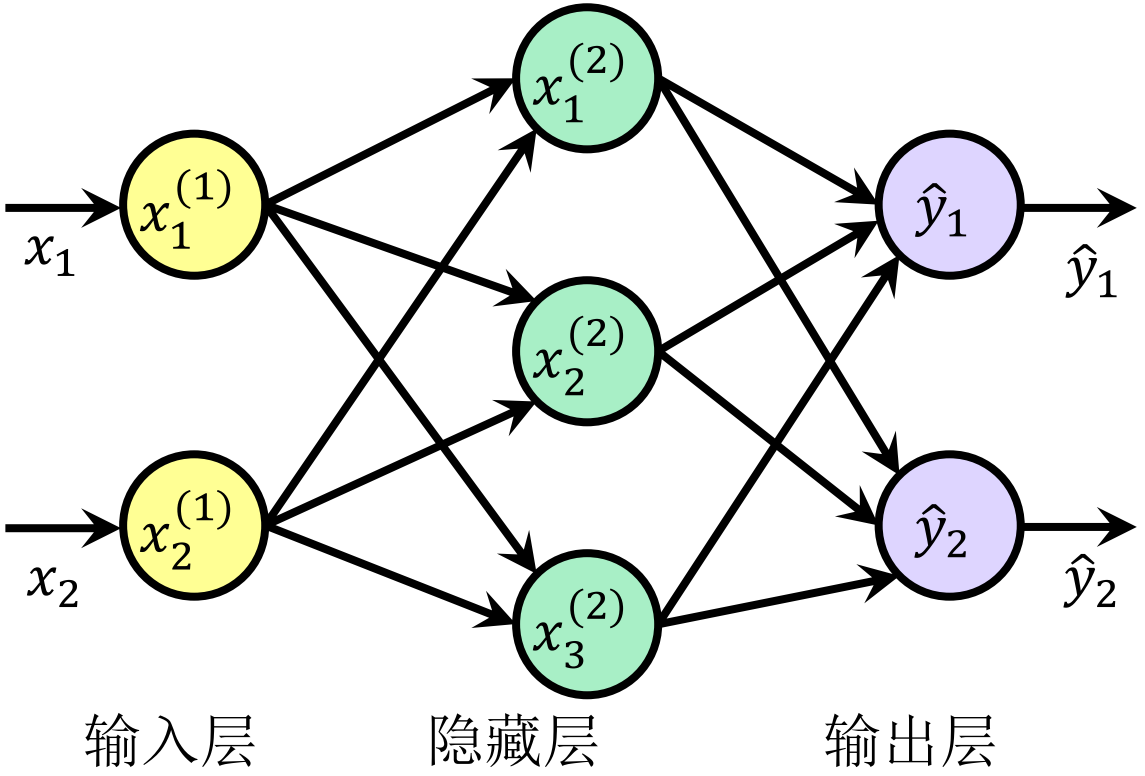
神经网络(Neural Network)

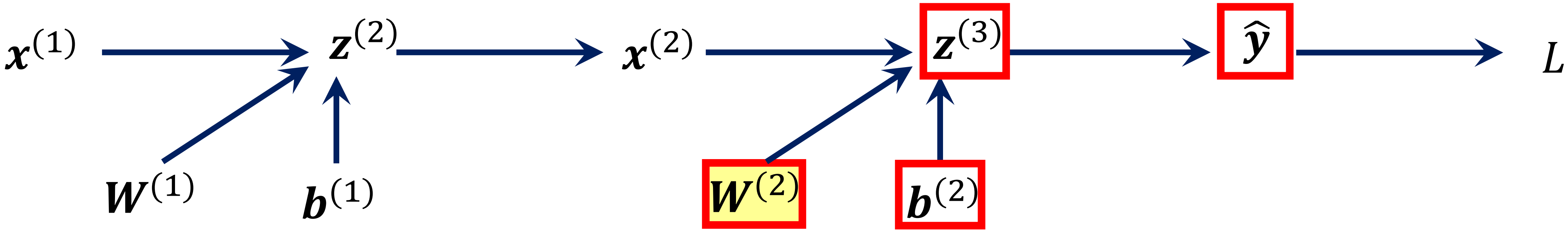
反向传播

$$\boldsymbol{z}^{(3)}=\boldsymbol{W}^{(2)}\boldsymbol{x}^{(2)}+\boldsymbol{b}^{(2)}$$

$$\frac{\partial L}{\partial \boldsymbol{W}^{(2)}}=\boxed{\frac{\partial \boldsymbol{z}^{(3)}}{\partial \boldsymbol{W}^{(2)}}}\frac{\partial L}{\partial \boldsymbol{z}^{(3)}}$$

无定义





神经网络(Neural Network)

$$\boldsymbol{z}^{(3)} = \boldsymbol{W}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{b}^{(2)}$$

为了表达简洁，省略上标，上式可以表示为

$$\boldsymbol{z} = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}$$

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} W_{11} & W_{12} & W_{13} \\ W_{21} & W_{22} & W_{23} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

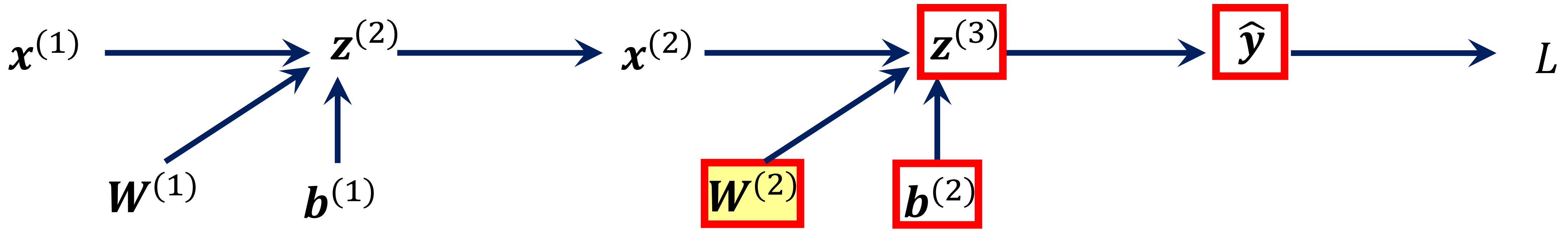
$$\frac{\partial L}{\partial W_{11}} = \frac{\partial L}{\partial z_1}\frac{\partial z_1}{\partial W_{11}} = \frac{\partial L}{\partial z_1}x_1, \quad \frac{\partial L}{\partial W_{12}} = \frac{\partial L}{\partial z_1}\frac{\partial z_1}{\partial W_{12}} = \frac{\partial L}{\partial z_1}x_2, \quad \frac{\partial L}{\partial W_{13}} = \frac{\partial L}{\partial z_1}\frac{\partial z_1}{\partial W_{13}} = \frac{\partial L}{\partial z_1}x_3$$

$$\frac{\partial L}{\partial W_{21}} = \frac{\partial L}{\partial z_2}\frac{\partial z_2}{\partial W_{21}} = \frac{\partial L}{\partial z_2}x_1, \quad \frac{\partial L}{\partial W_{22}} = \frac{\partial L}{\partial z_2}\frac{\partial z_2}{\partial W_{22}} = \frac{\partial L}{\partial z_2}x_2 \quad \frac{\partial L}{\partial W_{23}} = \frac{\partial L}{\partial z_2}\frac{\partial z_2}{\partial W_{23}} = \frac{\partial L}{\partial z_2}x_3$$

神经网络(Neural Network)

$$\frac{\partial L}{\partial \boldsymbol{W}}=\begin{bmatrix}\frac{\partial L}{\partial W_{11}} & \frac{\partial L}{\partial W_{12}} & \frac{\partial L}{\partial W_{13}}\\ \frac{\partial L}{\partial W_{21}} & \frac{\partial L}{\partial W_{22}} & \frac{\partial L}{\partial W_{23}}\end{bmatrix}=\begin{bmatrix}\frac{\partial L}{\partial z_1}x_1 & \frac{\partial L}{\partial z_1}x_2 & \frac{\partial L}{\partial z_1}x_3\\ \frac{\partial L}{\partial z_2}x_1 & \frac{\partial L}{\partial z_2}x_2 & \frac{\partial L}{\partial z_2}x_3\end{bmatrix}$$

$$=\begin{bmatrix}\frac{\partial L}{\partial z_1}\\ \frac{\partial L}{\partial z_2}\end{bmatrix}\begin{bmatrix}x_1 & x_2 & x_3\end{bmatrix}=\frac{\partial L}{\partial \boldsymbol{z}}\boldsymbol{x}^T$$

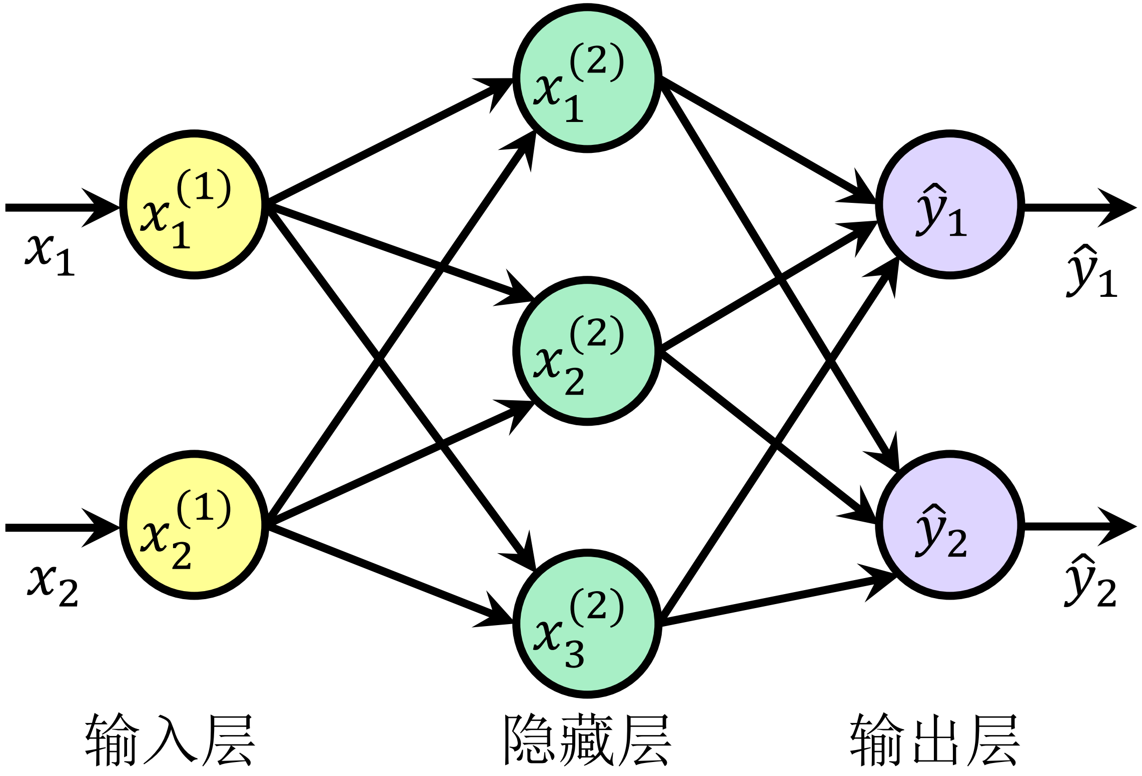


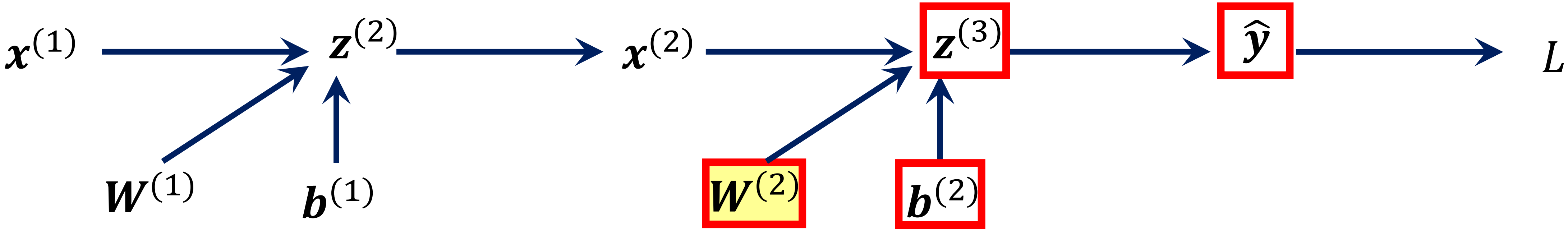
神经网络(Neural Network)

反向传播

$$\boldsymbol{z}^{(3)} = \boldsymbol{W}^{(2)}\boldsymbol{x}^{(2)} + \boldsymbol{b}^{(2)}$$

$$\frac{\partial L}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial L}{\partial \boldsymbol{z}^{(3)}} {\boldsymbol{x}^{(2)}}^{T}$$



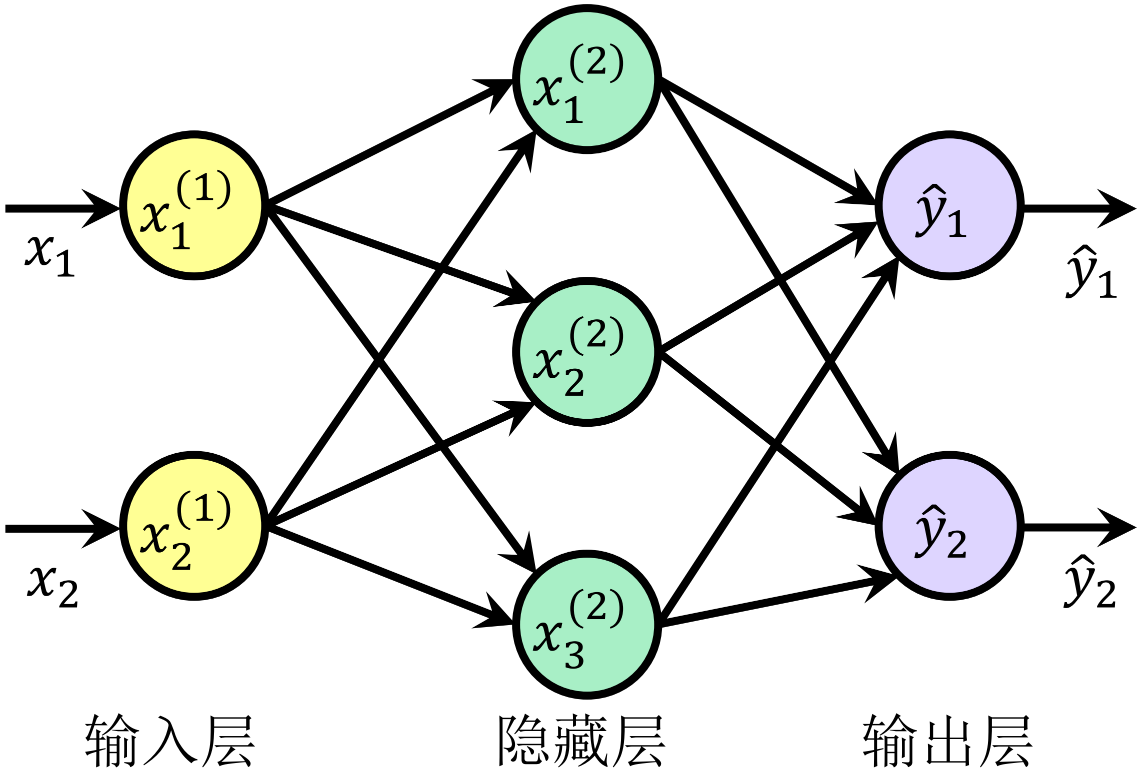


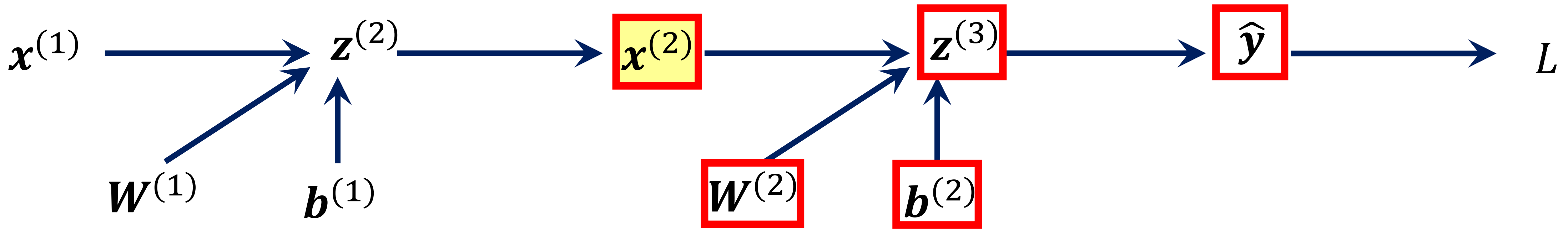
神经网络(Neural Network)

反向传播

$$\boldsymbol{z}^{(3)}=\boldsymbol{W}^{(2)}\boldsymbol{x}^{(2)}+\boldsymbol{b}^{(2)}$$

$$\frac{\partial L}{\partial \boldsymbol{x}^{(2)}}=\frac{\partial \boldsymbol{z}^{(3)}}{\partial \boldsymbol{x}^{(2)}}\frac{\partial L}{\partial \boldsymbol{z}^{(3)}}=\boldsymbol{W}^{(2)^T}\frac{\partial L}{\partial \boldsymbol{z}^{(3)}}$$



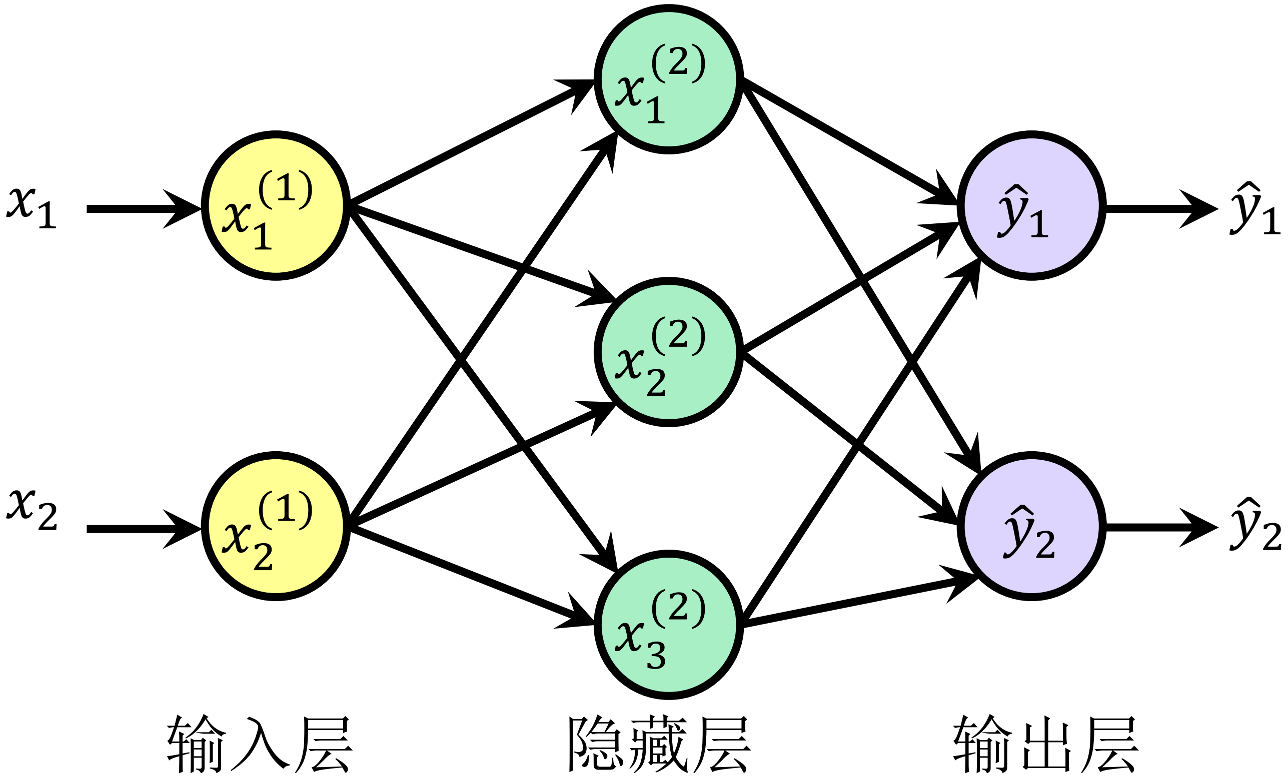


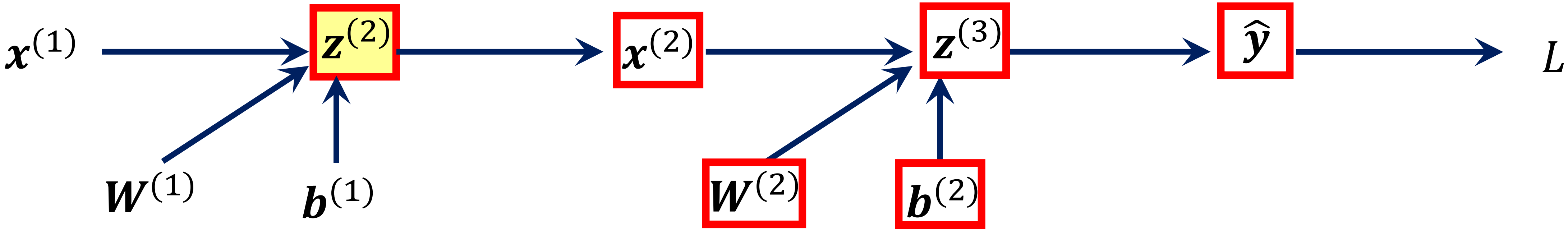
神经网络(Neural Network)

反向传播

$$\boldsymbol{x}^{(2)} = \sigma(\boldsymbol{z}^{(2)}) \quad \text{or} \quad \begin{cases} x_1^{(2)} = \sigma(z_1^{(2)}) \\ x_2^{(2)} = \sigma(z_2^{(2)}) \\ x_3^{(2)} = \sigma(z_3^{(2)}) \end{cases}$$

$$\frac{\partial L}{\partial \boldsymbol{z}^{(2)}} = \frac{\partial \boldsymbol{x}^{(2)}}{\partial \boldsymbol{z}^{(2)}} \frac{\partial L}{\partial \boldsymbol{x}^{(2)}}$$





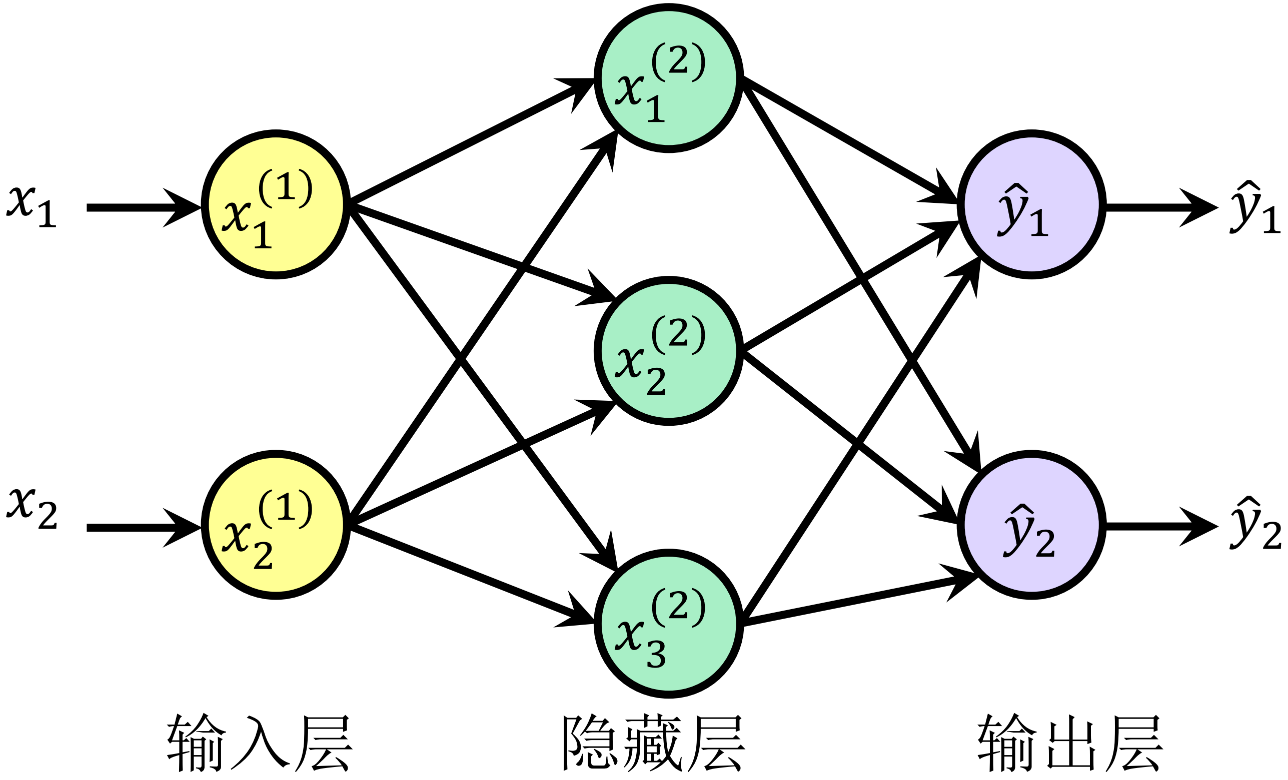
神经网络(Neural Network)

反向传播

$$\boldsymbol{x}^{(2)} = \sigma(\boldsymbol{z}^{(2)}) \qquad \begin{cases} x_1^{(2)} = \sigma(z_1^{(2)}) \\ x_2^{(2)} = \sigma(z_2^{(2)}) \\ x_3^{(2)} = \sigma(z_3^{(2)}) \end{cases}$$

$$\frac{\partial L}{\partial \boldsymbol{z}^{(2)}} = \frac{\partial \boldsymbol{x}^{(2)}}{\partial \boldsymbol{z}^{(2)}} \frac{\partial L}{\partial \boldsymbol{x}^{(2)}}$$

$$\frac{\partial \boldsymbol{x}^{(2)}}{\partial \boldsymbol{z}^{(2)}} = \begin{bmatrix} \sigma'(z_1^{(2)}) & 0 & 0 \\ 0 & \sigma'(z_2^{(2)}) & 0 \\ 0 & 0 & \sigma'(z_3^{(2)}) \end{bmatrix}$$

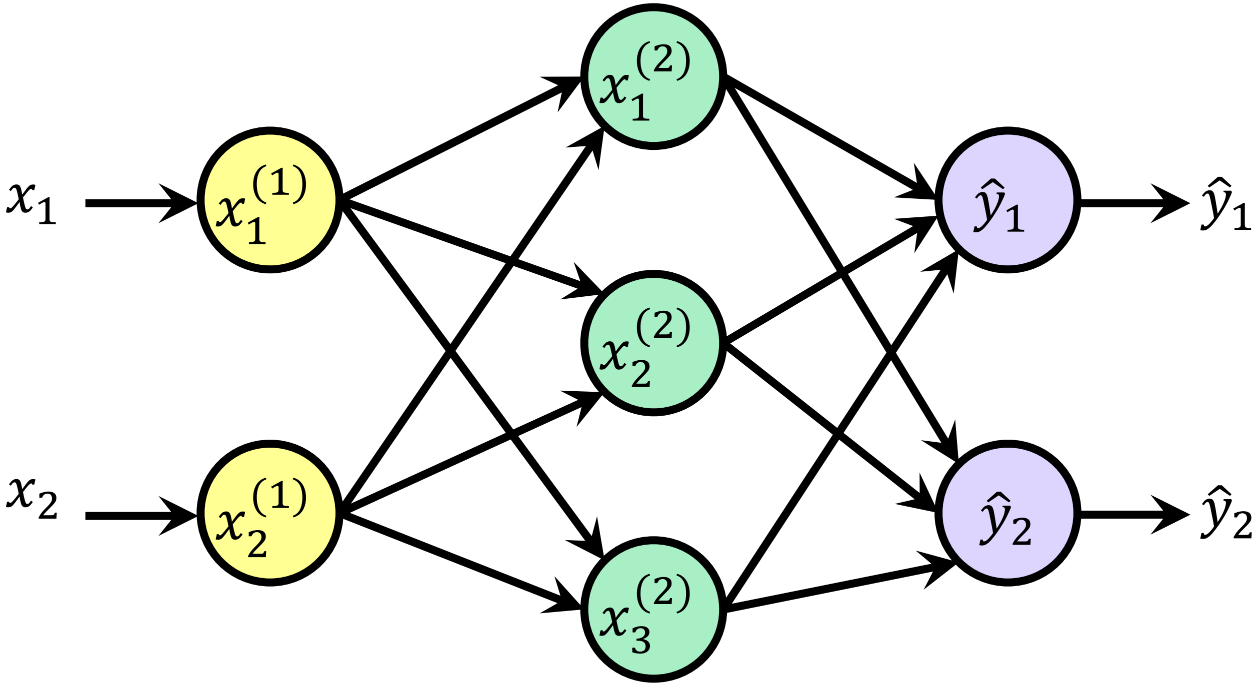


神经网络(Neural Network)

反向传播

$$\frac{\partial L}{\partial \boldsymbol{z}^{(2)}} = \frac{\partial \boldsymbol{x}^{(2)}}{\partial \boldsymbol{z}^{(2)}} \frac{\partial L}{\partial \boldsymbol{x}^{(2)}}$$

$$\frac{\partial \boldsymbol{x}^{(2)}}{\partial \boldsymbol{z}^{(2)}} = \begin{bmatrix} \sigma'(z_1^{(2)}) & 0 & 0 \\ 0 & \sigma'(z_2^{(2)}) & 0 \\ 0 & 0 & \sigma'(z_3^{(2)}) \end{bmatrix}$$

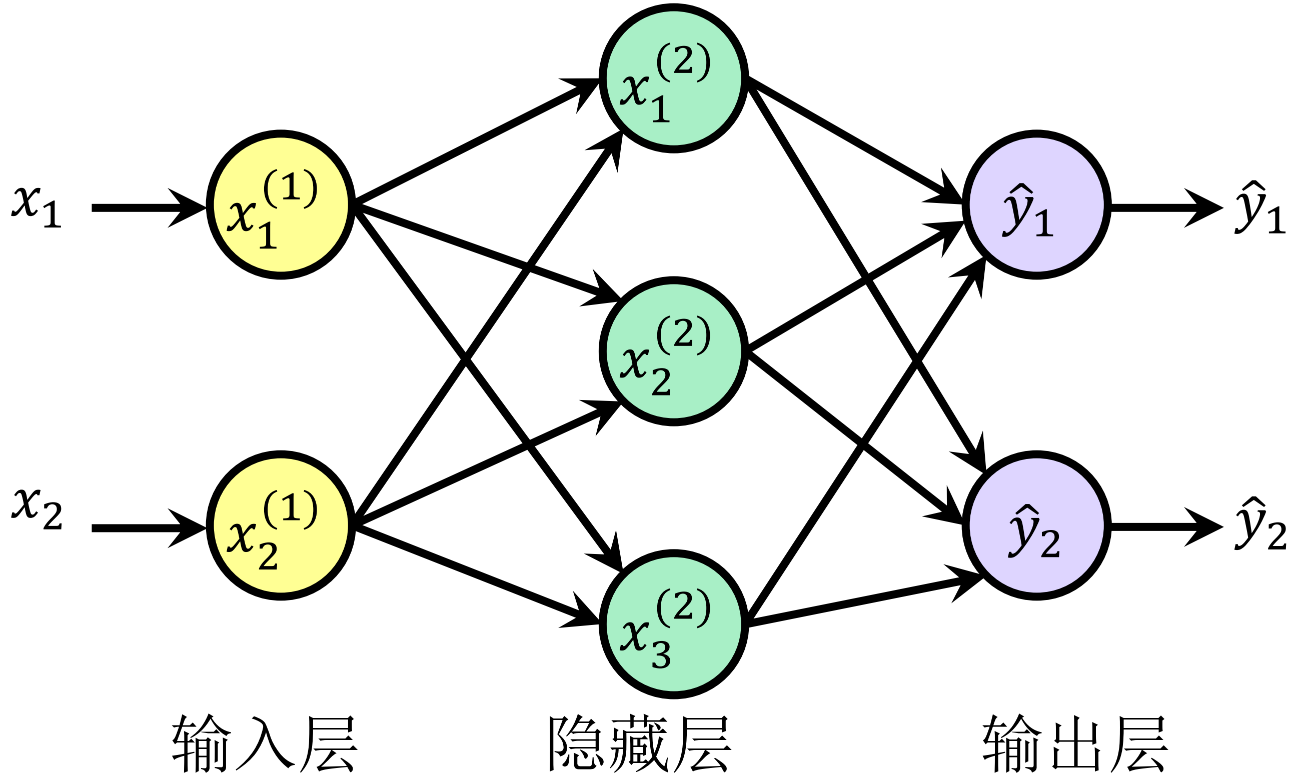


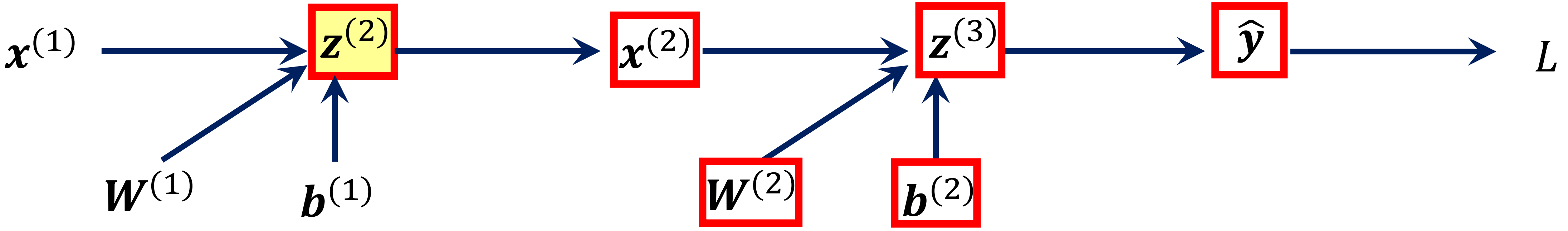
$$\frac{\partial L}{\partial \boldsymbol{z}^{(2)}} = \begin{bmatrix} \sigma'(z_1^{(2)}) & 0 & 0 \\ 0 & \sigma'(z_2^{(2)}) & 0 \\ 0 & 0 & \sigma'(z_3^{(2)}) \end{bmatrix} \begin{bmatrix} \frac{\partial L}{\partial x_1^{(2)}} \\ \frac{\partial L}{\partial x_1^{(2)}} \\ \frac{\partial L}{\partial x_1^{(2)}} \end{bmatrix} = \begin{bmatrix} \sigma'(z_1^{(2)}) \frac{\partial L}{\partial x_1^{(2)}} \\ \sigma'(z_2^{(2)}) \frac{\partial L}{\partial x_3^{(2)}} \\ \sigma'(z_3^{(2)}) \frac{\partial L}{\partial x_3^{(2)}} \end{bmatrix} = \begin{bmatrix} \sigma'(z_1^{(2)}) \\ \sigma'(z_2^{(2)}) \\ \sigma'(z_3^{(2)}) \end{bmatrix} \odot \begin{bmatrix} \frac{\partial L}{\partial x_1^{(2)}} \\ \frac{\partial L}{\partial x_3^{(2)}} \\ \frac{\partial L}{\partial x_3^{(2)}} \end{bmatrix} = \sigma'(\boldsymbol{z}^{(2)}) \odot \frac{\partial L}{\partial \boldsymbol{x}^{(2)}}$$

神经网络(Neural Network)

反向传播

$$\frac{\partial L}{\partial \boldsymbol{z}^{(2)}} = \frac{\partial \boldsymbol{x}^{(2)}}{\partial \boldsymbol{z}^{(2)}}\frac{\partial L}{\partial \boldsymbol{x}^{(2)}} = \sigma'(\boldsymbol{z}^{(2)})\odot\frac{\partial L}{\partial \boldsymbol{x}^{(2)}}$$





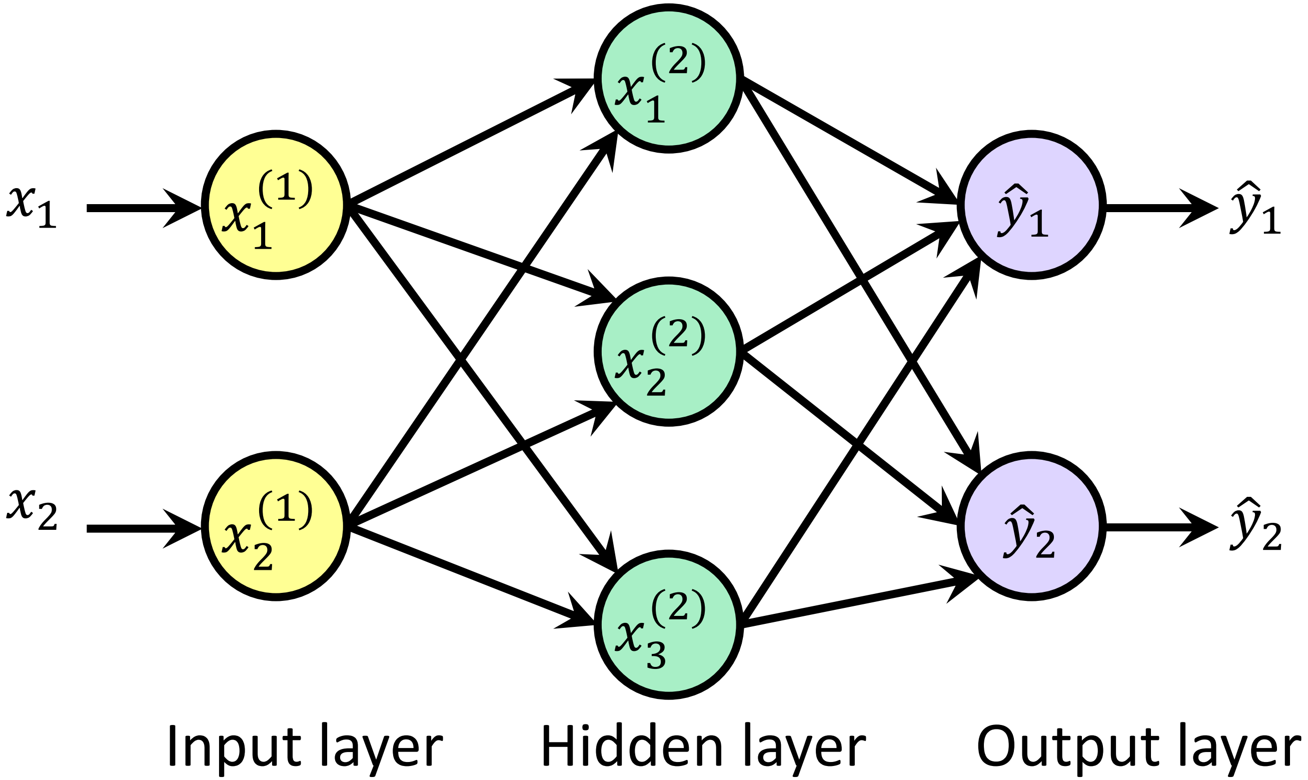
神经网络(Neural Network)

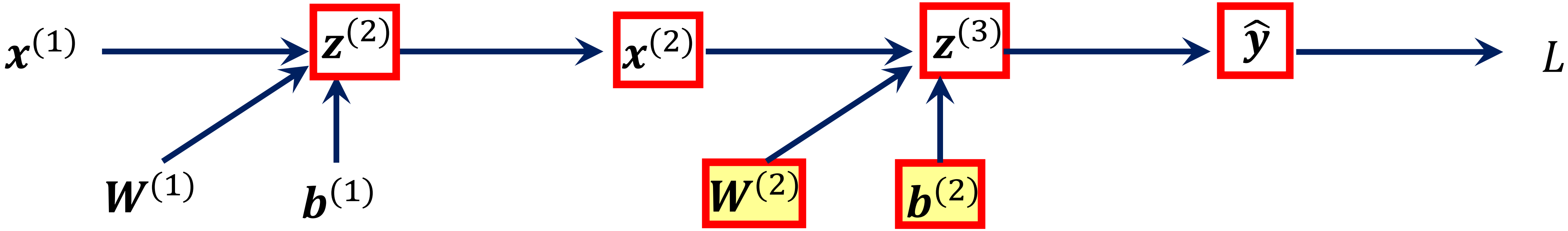
反向传播

$$\frac{\partial L}{\partial \boldsymbol{b}^{(2)}} = \frac{\partial \boldsymbol{z}^{(3)}}{\partial \boldsymbol{b}^{(2)}} \frac{\partial L}{\partial \boldsymbol{z}^{(3)}} = \boldsymbol{I} \frac{\partial L}{\partial \boldsymbol{z}^{(3)}}$$

$$\frac{\partial L}{\partial \boldsymbol{W}^{(2)}} = \frac{\partial L}{\partial \boldsymbol{z}^{(3)}} \boldsymbol{x}^{(2)^T}$$

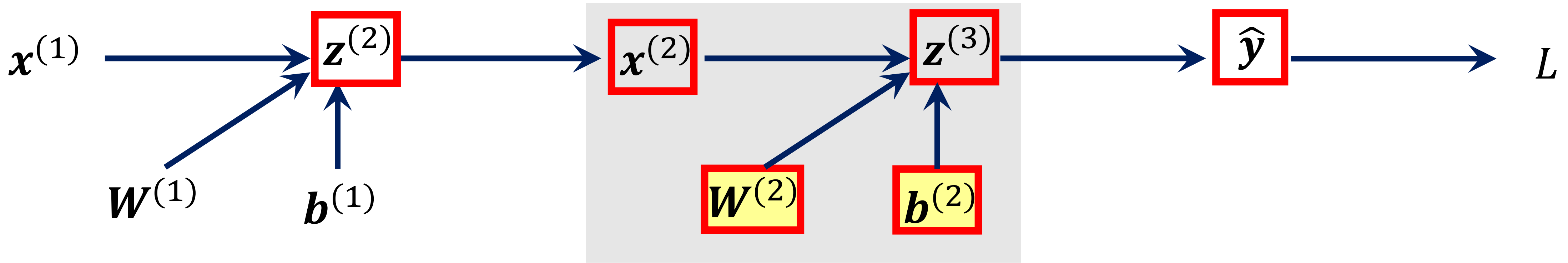
$$\frac{\partial L}{\partial \boldsymbol{x}^{(2)}} = \frac{\partial \boldsymbol{x}^{(2)}}{\partial \boldsymbol{z}^{(2)}} \frac{\partial L}{\partial \boldsymbol{x}^{(2)}} = \sigma'(\boldsymbol{z}^{(2)}) \odot \frac{\partial L}{\partial \boldsymbol{x}^{(2)}}$$





神经网络(Neural Network)

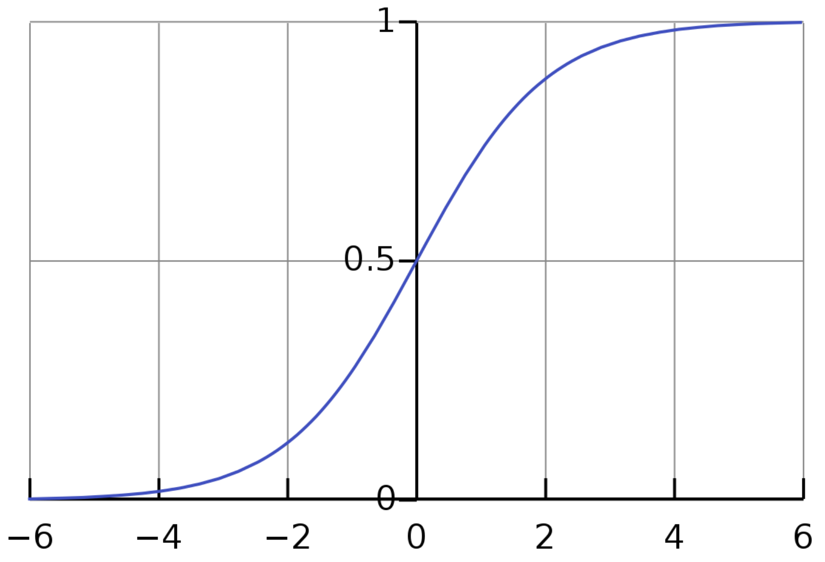
$$\begin{cases} \dfrac{\partial L}{\partial \boldsymbol{b}^{(2)}} = \dfrac{\partial L}{\partial \boldsymbol{z}^{(3)}} \\ \dfrac{\partial L}{\partial \boldsymbol{W}^{(2)}} = \dfrac{\partial L}{\partial \boldsymbol{z}^{(3)}} {\boldsymbol{x}^{(2)}}^T \end{cases} \Rightarrow \begin{cases} \dfrac{\partial L}{\partial \boldsymbol{b}^{(1)}} = \dfrac{\partial L}{\partial \boldsymbol{z}^{(2)}} \\ \dfrac{\partial L}{\partial \boldsymbol{W}^{(1)}} = \dfrac{\partial L}{\partial \boldsymbol{z}^{(2)}} {\boldsymbol{x}^{(1)}}^T \end{cases}$$

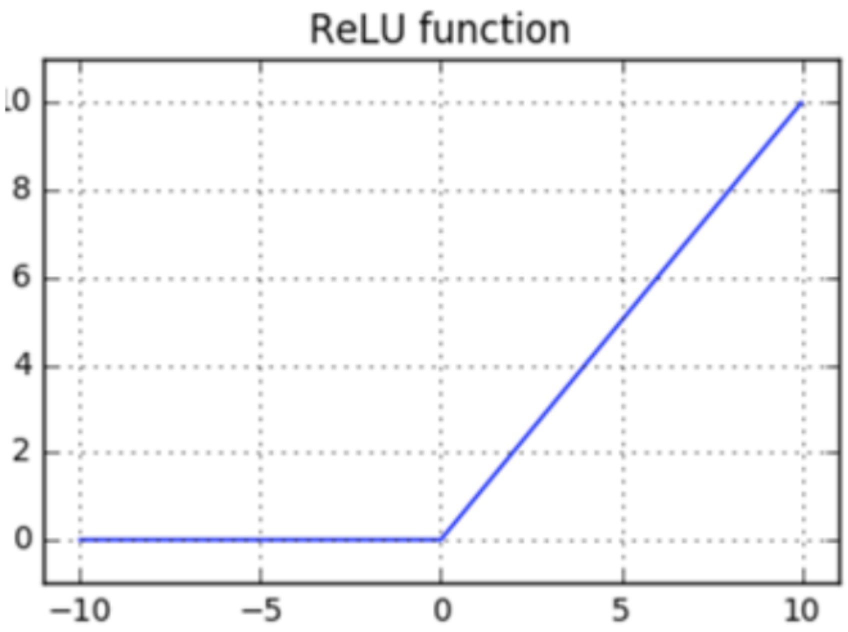


神经网络(Neural Network)

反向传播

- 问题: 梯度消失
- 解决方案： ReLU 函数(rectified linear unit, ReLU)



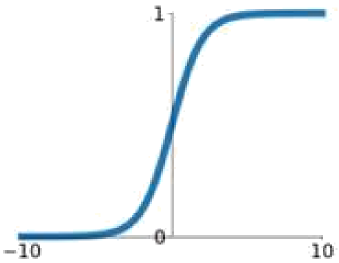


神经网络(Neural Network)

非线性激活函数

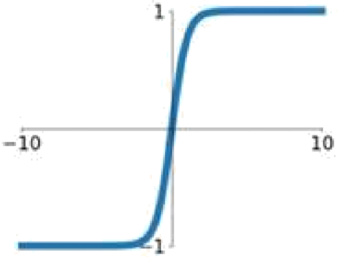
Sigmoid

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



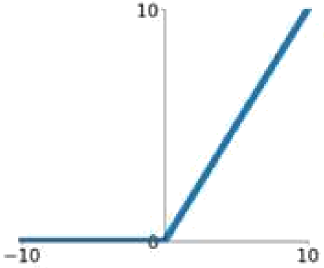
tanh

$\tanh(x)$



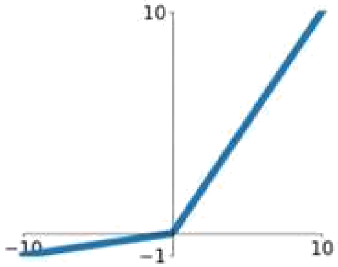
ReLU

$\max(0, x)$



Leaky ReLU

$\max(0.1x, x)$

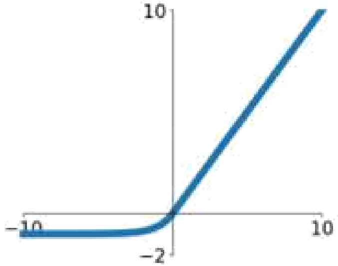


Maxout

$\max(w_1^T x + b_1, w_2^T x + b_2)$

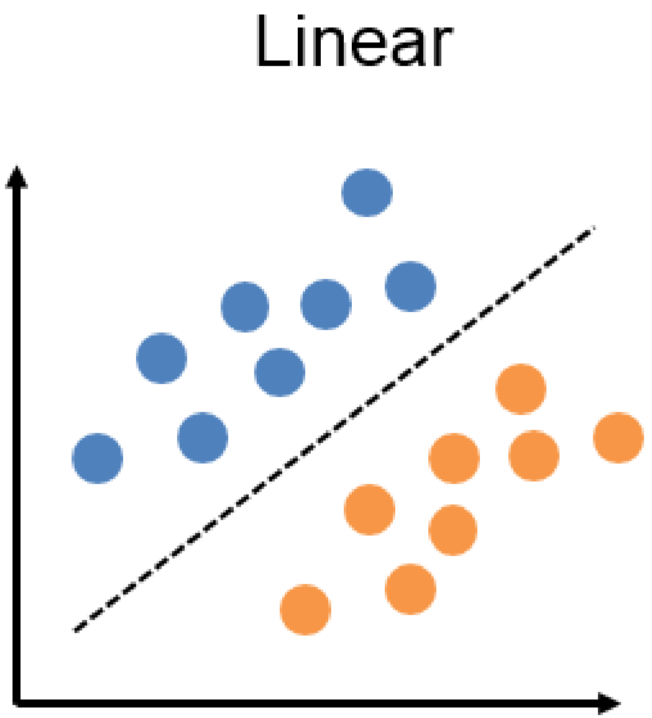
ELU

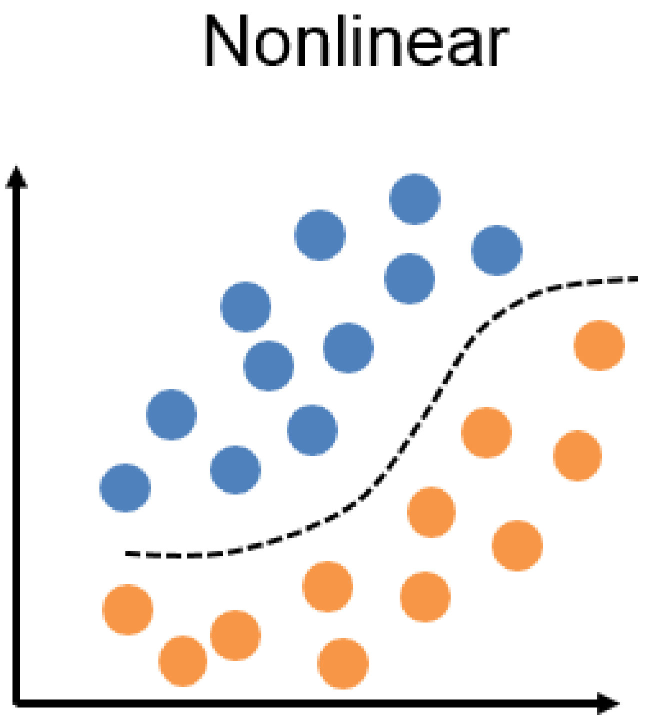
$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



神经网络(Neural Network)

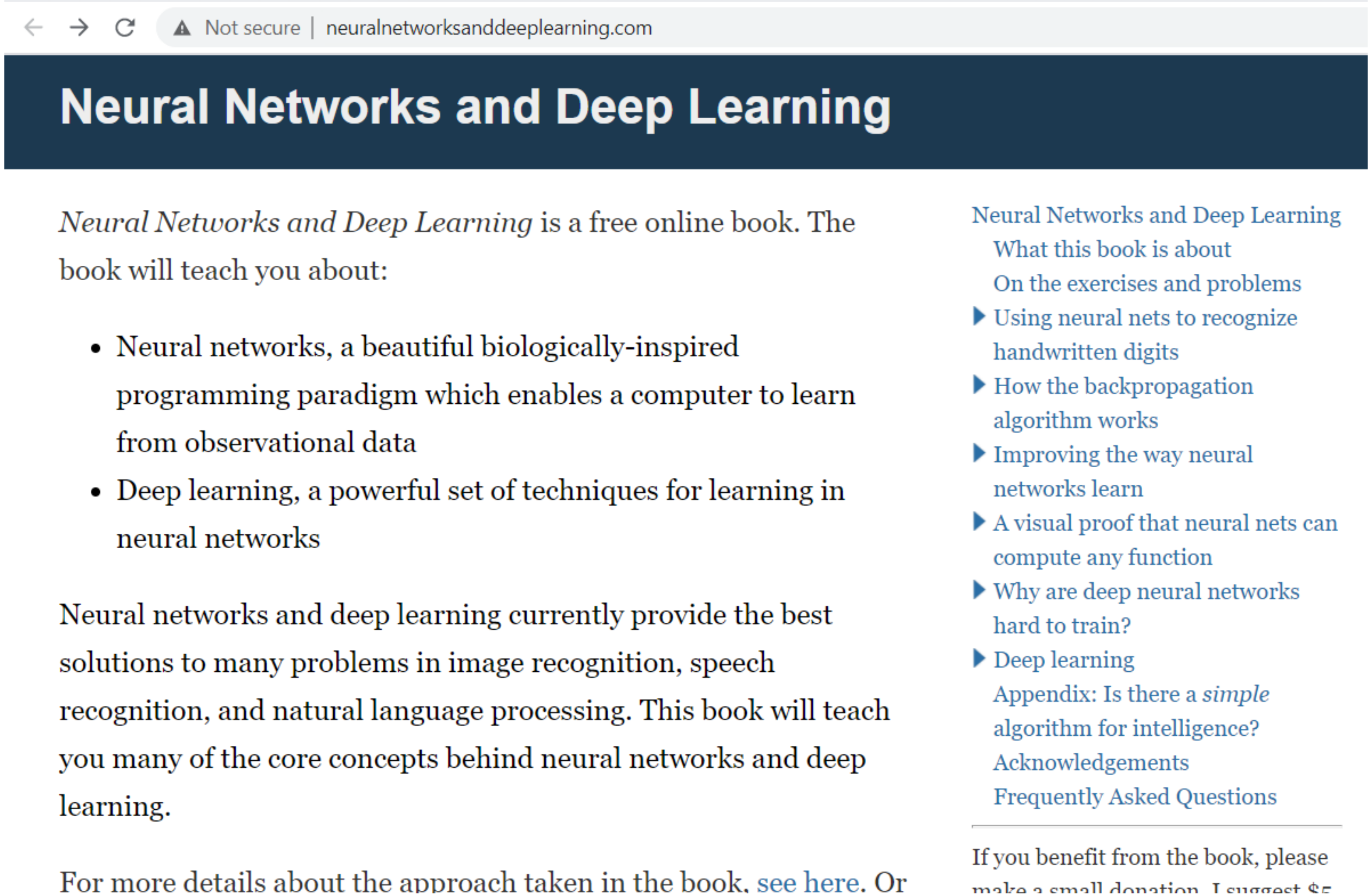
神经网络是一个非线性分类器





神经网络(Neural Network)

- Michael Nielsen , Neural network and deep learning, http://neuralnetworksanddeeplearning.com/



Not secure | neuralnetworksanddeeplearning.com

Neural Networks and Deep Learning

Neural Networks and Deep Learning is a free online book. The book will teach you about:

- Neural networks, a beautiful biologically-inspired programming paradigm which enables a computer to learn from observational data
- Deep learning, a powerful set of techniques for learning in neural networks

Neural networks and deep learning currently provide the best solutions to many problems in image recognition, speech recognition, and natural language processing. This book will teach you many of the core concepts behind neural networks and deep learning.

For more details about the approach taken in the book, see here. Or

If you benefit from the book, please

Neural network

- **QUESTION:** why do we need nonlinear activation function?