

Week 05.

5.1 神经元模型.

$$y_i = f(\sum_i w_i x_i - \theta_i)$$

其中 f 为 激活函数

5.2 感知机

与: $y = f(x_1 + x_2 - 2)$

或: $y = f(x_1 + x_2 - 0.5)$

非: $y = f(-0.6x_1 + 0.5)$

调整原则: $w_i \leftarrow w_i + \Delta w_i$

$$\Delta w_i = \eta (y - \hat{y}) x_i$$

5.3 BP 算法.

$$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\} \quad x_i \in \mathbb{R}^d, y_i \in \mathbb{R}^1$$

$$\alpha_k = \sum_{i=1}^d w_{ik} x_i, \quad b_k = \sigma(\alpha_k)$$

$$B_j = \sum_{i=1}^q w_{kj} x_j, \quad \hat{y} = \sigma(B_j)$$

σ : sigmoid 函数.

$\alpha_k \Rightarrow$ 第 k 个隐层的输入, $B_j \Rightarrow$ 第 j 个输出层的输入, $b_k \Rightarrow$ 第 k 个隐层的输出

对于 (x_k, y_k) 此时 $E_k = \frac{1}{2} \sum_{j=1}^l (\hat{y}_j^k - y_j^k)^2$

参数: $d \times q, q \times l, d, q$

$$v \leftarrow v - \eta \Delta v$$

$$\Delta w_{kj} = \frac{\partial E_k}{\partial w_{kj}} = \frac{\partial E_k}{\partial \hat{y}_j^k} \cdot \frac{\partial \hat{y}_j^k}{\partial B_j} \cdot \frac{\partial B_j}{\partial w_{kj}} = (\hat{y} - y) \cdot \hat{y} (1 - \hat{y}) x_j$$

缓解 BP 网 过拟合 $\left\{ \begin{array}{l} \text{早停: 训练, 验证, } E_{\text{train}} \text{ 下降, } E_{\text{valid}} \text{ 上升 则停止} \\ \text{正则化: } E = \lambda \frac{1}{n} \sum_{k=1}^m E_k + (1 - \lambda) \sum_i w_i^2 \end{array} \right.$

5.4. 跳出局部极小的方法

- ① 不同初始点
- ② 模拟退火 : 求多次最优解
- ③ SGD
- ④ 遗传算法.

5.5 其它常见网络.

(1) RBF

输入: d 维 x , 输出: 实值

$$p(x) = \sum_{i=1}^q w_i p(x, c_i)$$

$$p(x, c_i) = e^{-\beta_i \|x - c_i\|^2}$$

- 训练方法:
1. 采样聚类 $\rightarrow c_i$
 2. BP 算法 $\rightarrow w_i, \beta_i$

(3) SOM: 自组织映射

高维 \Rightarrow 低维, 保持拓扑结构.

输入 \rightarrow 计算最佳匹配单元 \rightarrow 更新权重 至收敛.

(6) Boltzmann 机

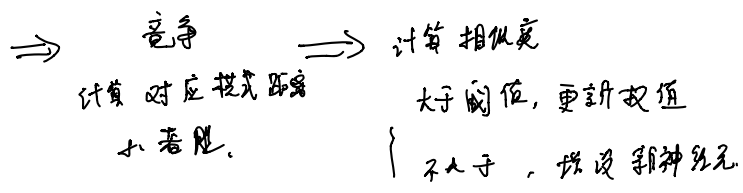
最小化能量 $E(s) = - \sum_{i=1}^{n-1} \sum_{j=i+1}^n w_{ij} s_i s_j - \sum_{i=1}^n \theta_i s_i$

神经元状态分布: $p(s) = \frac{e^{-E(s)}}{\sum_t e^{-E(s_t)}}$

受限 Boltzmann 机: 对比散度训练

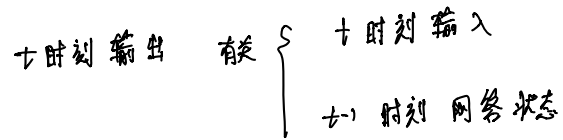
(2) ART: 自适应谐振网络,

比较层, 识别层, 识别层阈值, 重置模块.



(4) 级联网络: 结构自适应

(5) Elman 网络: 递归网络



深度学习

无监督逐层训练:

1. 预训练: 上层输出作下层输入

2. BP 训练

或 CNN 权值共享

$$p(v|h) = \prod_{i=1}^d p(v_i|h_i), \quad p(h|v) = \prod_{j=1}^q p(h_j|v_j)$$

互相迭代 $\Delta w = n(vh^T - v'h'^T)$

w_1 ↘
 w_2 ↘
 x_1 ↗
 x_2 ↗
 b ↗

$$z = w_1 x_1 + w_2 x_2 + b \longrightarrow \hat{y} = \sigma(z) \longrightarrow \underline{L(\hat{y}, y)}$$

$$- [y \log \hat{y} + (1-y) \log (1-\hat{y})]$$

$$\begin{aligned} \frac{\partial L}{\partial w_1} &= \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial z} \cdot \frac{\partial z}{\partial w_1} \\ &= \left[-\frac{y}{\hat{y}} + \frac{1-y}{1-\hat{y}} \right] \cdot \hat{y}(1-\hat{y}) \cdot x_1 \\ &= (\hat{y} - y) x_1 \end{aligned}$$

$$\frac{\partial L}{\partial w_2} = (\hat{y} - y) x_2, \quad \frac{\partial L}{\partial b} = \hat{y} - y$$

$$dw = 0, \quad db = 0$$

$$z = \text{np.dot}(w^T, x) + b \Rightarrow \begin{cases} w^T & (1, n_x) \\ x & (n_x, m) \end{cases}$$

$$\hat{y} = 1 / (1 + \text{np.exp}(-z)) \quad b \text{ 使用广播}$$

$$dz = \hat{y} - y \Rightarrow (1, m)$$

$$db = \frac{1}{m} \cdot \text{np.sum}(dz.T)$$

$$dw = \frac{1}{m} \cdot \text{np.sum}(\text{np.dot}(x, (dz).T))$$

$$\begin{aligned} db &: m, 1 \\ x &: n_x, m \\ dz &: 1, m \\ dw &: n_x, 1 \end{aligned}$$

$$w = w - \alpha dw$$

$$b = b - \alpha db$$

① $\text{np.random.randn}(5, 1)$ ✓
尽量不要用 (5)

② 用了请 $a.\text{reshape}(5, 1)$

③ 进一步确认

$\text{assert } a.\text{shape} == (5, 1)$