西瓜书复习笔记05

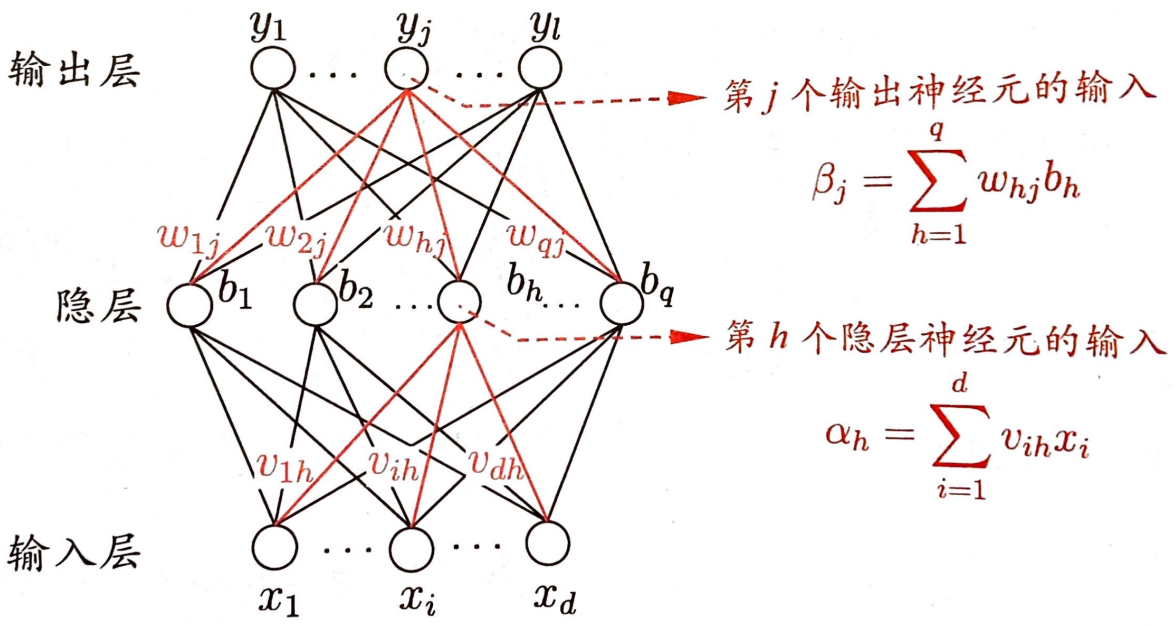
- **神经元模型：**
  - 激活函数：
    - 阶跃函数：
      理想的激活函数，对应输出值为0或1，其中0应对神经元抑制，1应对神经元兴奋
      缺点：不连续不光滑
    - Sigmoid：
      将较大范围的输出值挤压到$(0,1)$输出值范围内
  - bias：
    - 如果某神经元电位超过了bias，那么将被“激活”，向其他神经元发送化学物质
    - 如果某神经元电位低于bias，那么将被“抑制“
- **感知机与多层网络：**
  - 感知机：
    - 什么是感知机：
      感知机由两层神经元组成，输入层接受外界输入信号传给输出层，输出层是M-P神经元。
    - 什么是神经网络学习：
      就是根据训练数据，来调整神经元之间的连接权以及每个功能神经元的阈值。
    - 感知机权重更新：

$$w_i \leftarrow w_i + \Delta w_i$$

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

      其中
      η是学习率；
      若预测正确，感知机不发生变化，即$(y = \hat{y})$；
      若预测错误，感知机要进行权重调整$(y \neq \hat{y})$。
    - 收敛：
      感知机的学习过程使得权重向量趋于稳定。
    - 限制：
      感知机学习能力有限，只能解决与、或、非问题（线性问题）；
      但是不能解决像异或（非线性问题）。
    - 非线性可分：
      用线性超平面无法划分

    - 解决非线性问题：
      使用多层神经元，简单的两层感知机就能解决异或问题，即在输入层和输出层之间加入一层隐藏层（包含两个神经元，带激活函数）。
    - 多层前馈神经网络：
      每层神经元与上下一层神经元全互连，神经元之间不存在同层连接，也不存在跨层连接。
- **误差逆传播算法（反向传播，BP）：**
  - 什么是BP：
    - 从最后一层开始，利用广义的感知机学习规则，基于梯度下降策略，以目标的负梯度方向对参数进行调整。
  - BP的过程：
    1. 对每个训练样例，先将输入示例提供给输入层神经元，然后逐层将信号前传；
    2. 预测产生输出层结果，计算输出层误差；
    3. 将误差逆向传播至隐层，根据隐层神经元的误差来对连接权和阈值进行调整；
    4. 迭代循环进行，直到某些条件（如训练误差已经很小）停止。
  - BP算法的目的：
    - 不断的调节连接权，使最小化训练集上的累积均方误差。
  - BP算法权重更新：
    - 网络结构：



    - 神经网络输出：

$$\hat{y}_j^k = sigmoid\left(\beta_j - \theta_j\right)$$

      其中：
      θ_j表示阈值。
    - 网络上的均方误差：

$$E_k = \frac{1}{2}\sum_{j=1}^{l}\left(\hat{y}_j^k - y_j^k\right)^2$$

其中：

l表示输出层神经元的个数。

- 连接权更新：

$$w_{hj} \leftarrow w_{hj} + \Delta w_{hj}$$

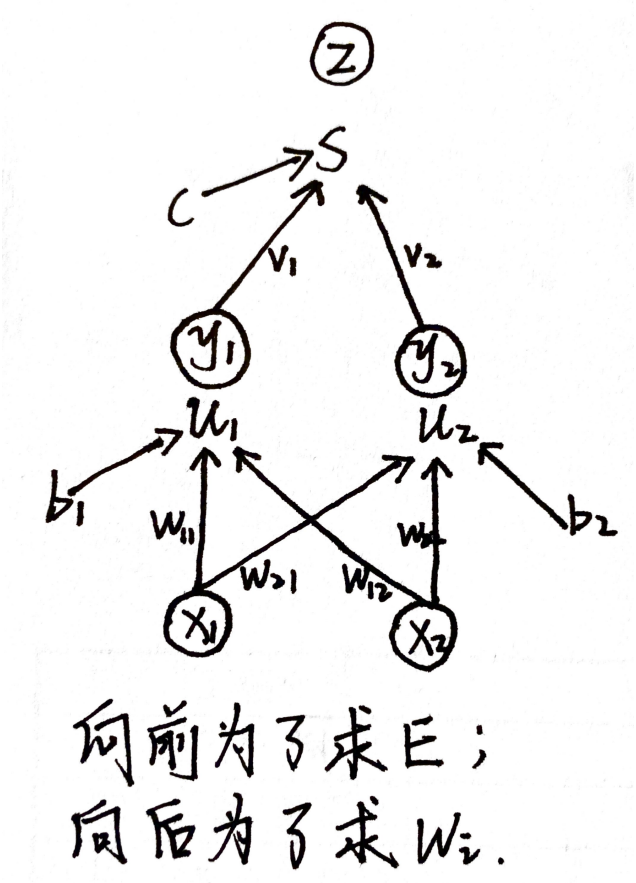
$$\Delta w_{hj} = -\eta \frac{\partial E_k}{\partial w_{hj}}$$

基于梯度下降策略，以目标函数的负梯度方向对参数进行调整。

- 链式法则（chain rule）：

$$\frac{\partial E_k}{\partial w_{hj}} = \frac{\partial E_k}{\partial \hat{y}_j^k} \cdot \frac{\partial \hat{y}_j^k}{\partial \beta_j} \cdot \frac{\partial \beta_j}{\partial w_{hj}}$$

- 例题：



$$u_1 = w_{11}x_1 + w_{12}x_2 + b_1$$

$$y_1 = g(u_1)$$

$$s = v_1 y_1 + v_2 y_2 + c$$

$$z = g(s)$$

$$E = \frac{1}{2}\sum (z-t)^2$$

$$\frac{\partial E}{\partial w_{11}} = \frac{\partial E}{\partial z} \cdot \frac{\partial z}{\partial s} \cdot \frac{\partial s}{\partial y_1} \cdot \frac{\partial y_1}{\partial u_1} \cdot \frac{\partial u_1}{\partial w_{11}}$$

$$\frac{\partial E}{\partial z} = z - t$$

$$\frac{\partial z}{\partial s} = z \cdot (z-t)$$

$$\frac{\partial s}{\partial y_1} = v_1$$

$$\frac{\partial y_1}{\partial u_1} = y_1 \cdot (1-y_1)$$

$$\frac{\partial u_1}{\partial w_{11}} = x_1$$

- sigmoid函数：
  - 每个隐藏层和输出层之间都有一个sigmoid函数。
  - sigmoid一个很好的性质：

$$f'(x) = f(x)(1-f(x))$$

    - 学习率：
        - 学习率太大容易震荡，太小收敛速度慢
          （具体参考西瓜书复习笔记02）
    - One Epoch:
        - 读取训练集一遍称为进行了一轮学习
    - 标准BP算法：
        - 每次更新只针对于单个样例，参数更新频率高（随机梯度下降）
        - 缺点：不同样例之间出现抵消现象
    - 累积BP算法：
        - 读取完整个训练集D后，才对参数进行更新，参数更新频率低（批量随机下降）
        - 缺点：累积误差下降到一定程度后，进一步下降速度慢
    - 过拟合怎么处理：
        - 正则化
        - 早停：
          训练集误差降低，验证集误差增高，停止；或达到预定轮数停止。
        - dropout
- 局部最小与全局最小：
    - 神经网络训练：
      可以看作是一个参数寻优的过程，即在参数空间中，找寻一组参数使得E最小。
    - 局部最小和全局最小的关系：
        - 局部最小对应参数产生的误差函数值大于全局最小对应的误差函数值。
    - 如何跳出局部最小：
        - 初始化参数：
          使用多种不同的参数，训练多个网络，取其中误差最小的解作为最终参数。
        - 模拟退火：
          模拟退火在每一步都以一定概率接受比当前解更差的结果，从而有助于跳出局部极小，在每部迭代过程中，接受次优解的概率随着时间的推移而逐渐降低，从而保证算法稳定。
        - 随机梯度下降：
          就算找到了局部最小点，它的梯度也不为零，有机会跳出局部最小。
        - *遗传算法：
          多个极小值竞争。
- 其他神经网络：
    - BRF网络：
      是一个单隐藏层前馈网络，径向奇函数作为隐藏层激活函数，输出层是隐藏层神经元的线性组合
    - ART网络：
      使用竞争型学习（无监督学习，网络的输出神经元相互竞争，每层仅激活一个神经元），比较层负责控制输入，识别层每个神经元对应一个模式类（神经元可动态添加）。
      训练时，识别层进行距离度量学习，对一个输入距离近的识别层神经元抑制其他神经元的激

活，距离大于识别阈值则将输入归入识别层神经元的模式，同时更新网络连接权。识别过程中动态调整神经元个数以适应输入模式的变化。

- SMO网络：
  自组织映射网络，将高维输入数据映射到低维空间（通常为二维），同时保持输入数据在高维空间的拓扑结构，即将高位空间中相似的样本点映射到网络输出层的临近神经元。
  训练时，接收一个训练样本，每个输出层神经元计算该样本与自身携带的权向量之间的距离，距离最近的神经元成为竞争获胜者，称为最佳匹配单元。
- 级联相关网络：
  级联相关网络有两个主要成分：级联和相关，级联是指建立层次连接的层级结构，在开始训练时， 网络只有输入层和输出层，处于最小拓扑结构，随着训练的进行，新的因曾神经元逐渐加入，从而 创建起层级结构，当新的隐层神经元加入时，其输入端连接权值是冻结固定的。相关是指通过最大
- Elman网络：
  最常用的递归神经网络之一，结构与多层前馈网络相似，但隐层神经元的输出被反馈回来，与下一时刻输入层神经元提供的信号一起作为隐层神经元在下一时刻的输入。
- Boltzmann机：
  通常分两层：显层与隐层，显层用于表示数据的输入与输出，隐层则被理解为数据的内在表达，神经元是布尔型的，只能取0、1两种状态，状态1表示激活，状态0表示抑制，状态向量出现的概率将仅由其能量与所有可能状态向量的能量确定。
  训练过程就是将每个训练样本视为一个状态向量，使其出现的概率尽可能大。
- 受限Boltzmann机：
  标准Boltzmann机是一个全连接图，训练网络的复杂度很高，这使其难以用于解决现实任务，现实中常用受限Boltzmann机仅保留显层与隐层之间的连接，从而将Boltzmann机结构由完全图简化为二部图。

- **交叉熵和kl散度：**
  https://github.com/yhangf/ML-NOTE/blob/master/pdf/多分类问题的交叉熵.pdf
  https://blog.csdn.net/Dby_freedom/article/details/83374650
- **在线学习与离线学习：**
  - 在线学习（online learning）：
    一个数据点训练完了直接更新权重（而不是一个batch）。
  - 离线学习（batch learning）：
    一个batch训练完才更新权重，这样的话要求所有的数据必须在每一个训练操作中（batch中）都是可用的。
  - 什么工业界模型的在线学习：
    传统的训练方法，模型上线后，更新的周期会比较长（一般是一天，效率高的时候为一小时），这种模型上线后，一般是静态的（一段时间内不会改变），不会与线上的状况有任何互动，假设预测错了，只能在下一次更新的时候完成更正。Online Learning训练方法不同，会根据线上预测的结果动态调整模型。如果模型预测错误，会及时做出修正。因此，Online Learning能够更加及时地反映线上变化。