

神经网络大作业

July 12, 2023

Contents

1	神经网络搭建	1
1.1	模型构建	1
1.2	任务分析	2
2	结果分析	2

1 神经网络搭建

1.1 模型构建

我们记第一层参数 $x1 \sim x4$, 中间连接参数为 v_{ij} , 第二层为 $s1 \sim s4$, 中间参数为 $w1 \sim w4$ 最后输出结果为 y 。整体神经网络数学模型如下所示：

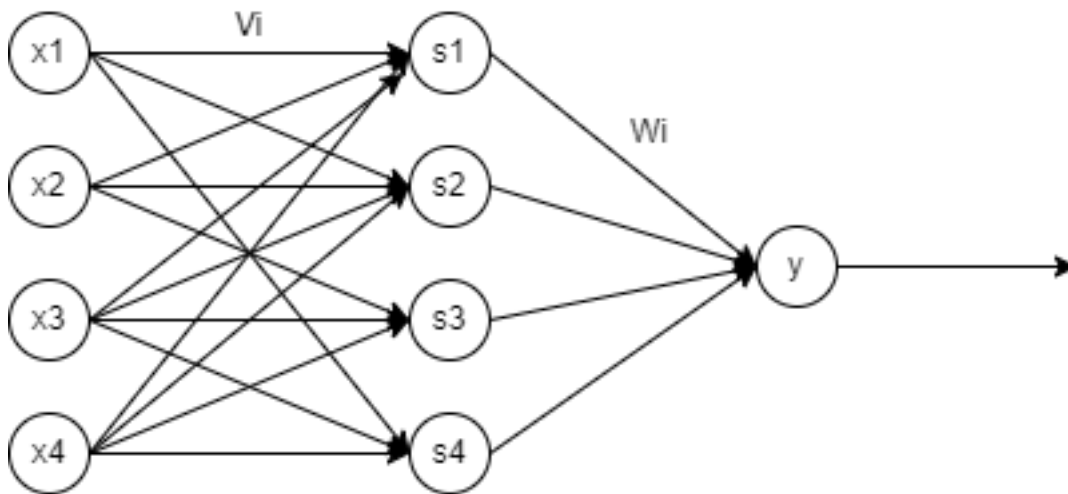


Figure 1: 感知机

数学公式说明：

$$\begin{aligned}
 s_j &= \text{sig}\left(\sum_{i=1}^4 x_i v_{ij} + b_j\right) \\
 y &= \text{sig}\left(\sum_{i=1}^4 s_i w_i + b\right) \\
 \text{loss} &= \frac{1}{2}(y - \text{res})^2 \\
 \frac{\partial \text{loss}}{\partial w_i} &= (y - \text{res}) \frac{\partial y}{\partial w_i} \\
 \frac{\partial y}{\partial w_i} &= \text{sig}(f)(1 - \text{sig}(f)) \frac{\partial f}{\partial w_i} = y(1 - y) \frac{\partial f}{\partial w_i} = y(1 - y) s_i \\
 \frac{\partial f}{\partial v_{ij}} &= \frac{\partial f}{\partial s_j} \frac{\partial s_j}{\partial v_{ij}} = w_j * \text{sig}(g)(1 - \text{sig}(g)) \frac{\partial g}{\partial v_{ij}} = w_j * s_j(1 - s_j) x_i \\
 \frac{\partial y}{\partial b} &= y(1 - y) \\
 \frac{\partial f}{\partial b_i} &= w_i * s_i(1 - s_i)
 \end{aligned} \tag{1}$$

这样完成我们的二层感知机构建

1.2 任务分析

我们需要将数据以 $[0,0,0,0],[1,1,0,1]$ 这样的方式传入网络第一层中（一共 16 个），它对应的理论值由它里面 ‘1’ 的个数决定。然后经过训练反向传递后对我们的梯度进行更新。由于该网路较为简单，而分类任务比较困难，因此很容易出现参数停滞不动的情况。特别是在经过实验之后我发现每一批数据直接取 16 会导致网络内参数变化率小、甚至出现长时间停滞的情况，因此我每一轮训练的时候一批数据取其中的随机 14 个，这样利于让参数跳出局部最优，更好的达到下降的目的。

对于第二问的动量法，我选取的公式为：

$$\Delta W(t) = \alpha \Delta W(t-1) - \eta \Delta E$$

当某一轮训练时，对于全部 16 种输入 x ，如果它对应的理论值与网络训练得到的结果误差都 ≤ 0.05 说明我们的网络训练到此结束。

2 结果分析

两问我们程序运行结果如下所示，其中得到的轮次是没有卡住的情况下运行轮次（也就是说在随机参数下网络训练一直停滞几乎不动的情况被我舍弃了）：

η	(1) 迭代次数	(2) 加动量迭代次数
0.05	48515	14160
0.10	25335	7074
0.15	15839	6137
0.2	12418	7789
0.25	16632	8950
0.3	11363	4757
0.35	9867	4832
0.4	5492	6056
0.45	6868	8311
0.5	5508	16547

我们可以看到随着 η 值升高，网络训练与收敛速度越来越快，整体迭代次数逐步下降。在加入动量机制后我们发现网络训练速度一下子就变得很快，比先前快了很多，但是在 η 很大时右会出现参数卡壳的情况，导致模型训练速度出现骤降。因此，选取合适的 η 以及优化器至关重要。